KU LEUVEN

ARENBERG DOCTORAL SCHOOL
Faculty of Engineering Science

Weakly Supervised Machine Learning Algorithms for Object Recognition in the Wild and Entity Linking in Videos

Aparna Nurani Venkitasubramanian

Supervisors:
Prof. dr. Marie-Francine Moens
Prof. dr. ir. Tinne Tuytelaars

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor of Engineering Science (PhD): Computer Science

June 2017

Weakly Supervised Machine Learning Algorithms for Object Recognition in the Wild and Entity Linking in Videos

Aparna NURANI VENKITASUBRAMANIAN

Examination committee:
Prof. dr. ir. Herman Neuckermans, chair
Prof. dr. Marie-Francine Moens, supervisor
Prof. dr. ir. Tinne Tuytelaars, supervisor
Prof. dr. Daniel De Schreye
Prof. dr. ir. Hendrik Blockeel
Prof. dr. Guillaume Gravier
(Institut de Recherche en Informatique et Systèmes Aléatoires (IRISA), France)
Prof. dr. Tatiana Tommasi
(Sapienza University of Rome, Italy)

Dissertation presented in partial fulfillment of the requirements for the degree of Doctor of Engineering Science (PhD): Computer Science

June 2017

Acknowledgements

I feel my PhD journey has been a highly rewarding one, and I am deeply indebted to the people who made this happen. First and foremost, I would like to thank my promoters and mentors Prof. Marie-Francine Moens and Prof. Tinne Tuytelaars for guiding me through the PhD.

Thank you, Sien - for introducing me to the exciting world of information retrieval, and giving me this wonderful opportunity. Your dedication and work ethic amaze me. You have inculcated so many qualities in me - attention to detail, hard work, systematic and methodical reasoning, for which I shall remain grateful. I acknowledge the fact that you made sure all administrative and visa related issues were well taken care of. Your prompt feedback even at short notice for all my papers is much appreciated.

Thank you, Tinne - You have taught me so many things beyond computer vision. You have always insisted on top-notch quality and raised the bar every single time. I can't thank you enough for staying up with me late into the nights, reviewing my work and ensuring that my work is understood by the CV community. Your emphasis on mathematical rigor has strengthened my understanding and the immense joy I gained thereafter has bolstered my self-confidence.

It is truly a privilege to have worked with you both. Words cannot do justice to my sense of gratitude. Hartelijk bedankt!

I would like to thank all the members of the examination committee: Prof. Danny De Schreye, Prof. Hendrik Blockeel, Prof. Guillaume Gravier and Prof. Tatiana Tomassi for agreeing to be on the Jury, evaluating this thesis, asking engaging questions, and providing valuable inputs to improve the quality of my work. I thank the chair Prof. Herman Neuckermans for his valuable time.

I remain indebted to all teachers from MAI program - Prof. Danny De Schreye, Prof. Hendrik Blockeel, Prof. Gerda Janssens, Prof. Luc de Raedt. The

coursework prepared me well to pursue a PhD and I strongly recommended MAI program to students interested in AI.

Thank you Geert and Tuur for all the brainstorming sessions and intellectually stimulating discussions. Alma lunches, Pomodoro breaks, quizzes, and games were so much fun. You made my stay so pleasant and meaningful. I wish you great success in your PhD.

I would like to extend my gratitude to the other fellow LIIRians- Quynh, Golnoosh, Susana, Niraj, Elias, Guillem, Katrien, Shurong, Oleksandr, Gert-Jan, Ivan, Jan, Karl, Phi and Wim for all their support and encouragement.

I would like to thank my friends at Leuven - Rebecca, Koen, Jyothsna, Santhosh, Parvathy, Sandeep, Asha, Arul, Divya, Praveen, Sathish, Sowmya, Gowri Shankar, Anjali, Ravi, Ranjini, Aparna, Nilesh, Durga, Hari for the warm company and for bringing a flavor of India here. A get-together is a perfect antidote for homesickness and I will cherish all the events.

My sincere gratitude to all my relatives - grandparents, aunts, uncles, cousins, mother-in-law, and sister-in-law. My special thanks to Raji perimma for all the pains she took to ensure a good career for me.

I would like to thank my parents for all the love and care they have showered on me, and the effort they took bringing me up. Thank you, Daddy, Mumma! You are the best parents in the whole world! You have always been with me, guided me towards important things in life, instilled high ethical and moral principles, provided continuous encouragement, and believed in me more than I ever did (will).

Thank you, Aravind, for being not only an awesome, protective and affectionate brother but also for being my best friend. Thank you for patiently listening to my endless rants, being my sounding board, and being there for me through thick and thin, no matter the geography, or the time zone differences, or how hectic your own schedule was. Thank you for prioritizing me over anything else.

Thank you, Kamesh, for being such a supportive husband, and for encouraging me to pursue my passion. Thank you for initiating me into the Masters in Artificial Intelligence program, and motivating me to take up a PhD in this field. Thank you for being actively involved in all my endeavors.

Thank you, Advaita, for being as understanding as one can be. You have taught me to find beauty in everyday things and refreshed me with your funny stories, songs, dances, and games. I hope you will read this someday, and feel happy you made my day, every day.

Abstract

With the proliferation of video-rich data on the Internet, there is a pressing need for search tools that can retrieve not only relevant videos from a corpus, but also relevant snippets within a video. For retrieving relevant videos, current search technologies hinge on labor-intensive manual annotation of tags, which are subjective and often incomplete. To fully automate search and retrieval systems, we need tools that can understand the content presented in videos and automatically generate labels that accurately describe them. Towards that goal, we consider a video with subtitles, and focus on two problems: a) What/ who are in the video key frames? and b) What do the textual entity mentions in subtitles refer to?

State-of-the-art methods largely adopt a supervised paradigm, relying on expensive manually created training examples to indicate the mapping between the visual and textual entities. In contrast, we address these questions using a weakly supervised paradigm, where the text may provide some clues on the vision and vice versa. We further apply it to the problem of wildlife recognition in nature documentaries.

In a weakly supervised setting, the problem of recognizing entities in vision and language presents a host of challenges for vision, text and the association of text and vision. On the vision side, we deal with a scenario where there are no visual demarcators to indicate the location of an animal. In fact, it is not even known if there are animals at all in a certain key frame. Additionally, since we are dealing with animals shot in their natural habitat, there are challenges due to self-occlusion, camouflage, illumination etc. On the textual side, while we have tools to detect entity mentions in the text, not all of them are pertinent to animals. Even when the mentions refer to animals, they are often so ambiguous that it is impossible to resolve them correctly without a holistic understanding of the context. As far as the linking of text and vision is concerned, the absence of visual demarcators in the visual data coupled with the presence of ambiguous mentions in text makes it harder to reliably tie together the entities in vision

and language. That is, there are no ready examples to show the association in a limited, diverse dataset.

In this thesis, we present three major contributions that address these challenges. First, we present a multi-modal domain adaptation framework for multi-label classification. Here, we propose an algorithm to learn from an external labeled source dataset, and iteratively adapt to a target dataset, by leveraging the weakly associated textual subtitles that come with the video. We prove that this approach is significantly better than a) a purely vision-based approach or b) purely text-based approach or c) an approach that uses both text and vision, but without labeled examples or d) an approach that uses both text and vision, and labeled (out-of-domain) examples, but without the adaptive learning.

Next, we investigate image representation and object recognition models learned from video documentaries by using the weak supervision of the textual subtitles. In particular, we study a support vector machine on top of activations of a pre-trained convolutional neural network, as well as a Naive Bayes framework on a *'bag-of-activations'* image representation, where each element of the bag is considered separately. On testing the models on a target dataset shot in entirely different conditions, we found that the *'bag-of-activations'* based model outperformed classical models by a huge margin.

The third and final contribution capitalizes on the inherent characteristics in the video, such as the temporal coherence in video frames, and the dependencies within and across the visual and textual modalities. We prove that this integrated modelling yields significantly better performance over text-based and vision-based approaches. We show that textual mentions that cannot be resolved using text-only methods are resolved correctly using our method.

The methods proposed here take us a step closer to object recognition in the wild and automatic video indexing. While the methods presented here have been validated on wildlife documentaries, they are all quite generic and can be applied to a plethora of other genres, beyond wildlife, beyond subtitles, or even beyond video documentaries.

Beknopte samenvatting

Door de snelle toename van videorijke data op het internet, is er een dringende nood aan zoekhulpmiddelen die niet enkel relevante video's uit een corpus kunnen terughalen, maar ook de relevante fragmenten binnen een video. Om relevante video's te vinden steunen de huidige zoektechnologieën op arbeidsintensieve, handmatige annotatie van labels, die subjectief en vaak onvolledig zijn. Om zoeksystemen volledige te automatiseren, hebben we hulpmiddelen nodig die de inhoud die voorgesteld wordt in video's kunnen begrijpen en automatisch labels kunnen generen die de video's accuraat beschrijven. Om naar deze doelstelling toe te werken, beschouwen we video's met ondertitels en focussen we op twee problemen: a) Wie/wat is er in de sleutelframes? en b) Naar waar verwijzen de tekstuele entiteit vermeldingen in ondertitels?

State-of-the-art methoden nemen grotendeels een gesuperviseerd paradigma aan, afhangend van kostbare, handmatig gecreëerde trainingsvoorbeelden om de relatie tussen visuele en tekstuele entiteiten aan te duiden. Daarentegen adresseren wij deze vragen met een zwak gesuperviseerd paradigma, waar de tekst enkele hints over het visie gedeelte kan bieden en vice versa. We passen dit paradigma ook toe op het probleem van wildlife herkenning in natuurdocumentaires.

In een zwak gesuperviseerde setting presenteert het herkennen van entiteiten in visie en taal een veelvoud aan uitdagingen voor visie, tekst en het linken van visie en tekst. Aan de visie kant, behandelen we een scenario waar er geen visuele demarcaties zijn die de locatie van een dier aangeven. Het is zelfs niet bekend of er überhaupt dieren aanwezig zijn in het key frame. Bovendien, mits we te maken hebben met dieren in hun natuurlijke omgeving, zijn er uitdagingen vanwege zelf-occlusie, camouflage, belichting, etc. Aan de tekstuele kant, hoewel we hulpmiddelen hebben om entiteit vermeldingen te detecteren, zijn deze niet allemaal van toepassingen op dieren. Zelfs wanneer de vermeldingen te maken hebben met dieren, zijn deze vaak zo ambigu dat het onmogelijk is deze correct op te lossen zonder een holistisch begrip van de context. Wat het linken van

tekst en visie betreft, maakt de afwezigheid van visuele demarcaties in de visuele data, gekoppeld aan de aanwezigheid van ambigue vermeldingen in de tekst, het moeilijker om de entiteiten op een betrouwbare manier te linken in visie en taal. D.w.z., er zijn geen voorbeelden beschikbaar om de associatie aan te geven in een beperkte, diverse dataset.

In deze thesis, presenteren we drie belangrijke contributies die deze uitdagingen adresseren. Ten eerste, presenteren we een multi-modaal domeinadaptatieframework voor multi-label classificatie. Hierin stellen we een algoritme voor om te leren vanuit een extern gelabelde bron-dataset en adapteren we iteratief naar een doel-dataset, door gebruik te maken van zwak-geassocieerde tekstuele ondertitels die bij de video horen. We bewijzen dat deze aanpak significant beter is dan a) een pure visie-gebaseerde aanpak, of b) een pure tekst-gebaseerde aanpak, of c) een aanpak die zowel visie als tekst gebruikt, maar zonder gelabelde voorbeelden, of d) een aanpak die zowel visie als tekst gebruikt en gelabelde (buiten-domein) voorbeelden, maar zonder het adaptief leren.

Vervolgens onderzoeken we beeldrepresentaties en objectherkenningsmodellen geleerd uit videodocumentaires door het gebruik van zwakke supervisie van de tekstuele ondertitels. We bestuderen met name een support vector machine bovenop activaties van een vooraf-getraind convolutioneel netwerk, als wel een naïef Bayes framework op een 'bag-of-words' representatie, waar elk element uit de 'bag' apart beschouwd wordt. Bij het testen van de modellen op een doel-dataset, opgenomen in volledig andere omstandigheden, vonden we dat het 'bag-of-activations'-gebaseerde model het klassieke modellen met een grote marge overtrof.

De derde en laatste contributie benadrukt de inherente karakteristieken in de video zoals de tijdscoherentie in de videoframes en de afhankelijkheden binnen en over de visuele en tekstuele modaliteiten heen. We bewijzen dat dit geïntegreerde modelleren significant betere performantie geeft over de tekst-gebaseerde en visie-gebaseerde aanpakken. We laten zien dat tekstuele vermeldingen die niet kunnen worden opgelost met methoden die alleen tekst gebruiken worden opgelost met behulp van onze methode.

List of Abbreviations

AI	Artificial Intelligence
ANN	Artificial Neural Network
API	Application Programming Interface
ASR	Automatic Speech Recognition
BBC	British Broadcasting Corporation
BoVW	Bag of Visual Words
BoW	Bag of Words
CEAF	Constrained Entity-Alignment F-Measure
CEAF_e	Constrained Entity-Alignment F-Measure for Entities
CEAF_m	Constrained Entity-Alignment F-Measure for Mentions
CNN	Convolutional Neural Network
CRF	Conditional Random Field
EM	Expectation-Maximization
GT	Ground Truth
JPD	Joint Probability Distribution
IP	Internet Protocol
LBP	Loopy Belief Propagation
MRF	Markov Random Field
MUC	Message Understanding Conference
NBC	Naive Bayes Classifier
NER	Named Entity Recognition
NEL	Named Entity Linking
NED	Named Entity Disambiguation
NERD	Named Entity Recognition and Disambiguation
NLP	Natural Language Processing
NN	Neural Network
POS	Parts-Of-Speech
SIFT	Scale-Invariant Feature Transform
SVM	Support Vector Machine
VIT	Viterbi Approximation

List of Symbols

β_j	The belief associated with clique $\mathbf{C}_j$
$\delta_{i \to j}$	The message from clique $\mathbf{C}_i$ to clique $\mathbf{C}_j$
μ	Mean of a normal distribution
μ_n	Mean associated with class n for a normal distribution
π_i	The parents of node X_i
ψ_i	The potential function associated with node i
ψ_{text}	The potential function associated with textual nodes
ψ_{text_vision}	The potential function associated with the edges across textual and visual nodes
ψ_{vision}	The potential function associated with visual nodes
σ^2	Variance of a normal distribution
σ_n^2	Variance associated with class n for a normal distribution
Θ	A set of parameters
θ_l	The set of parameters governing the label y_l
$\mathbf{A}$	A set of feature vectors, each one representing a key frame
$\mathbf{a_i}$	Feature vector denoting i^{th} video key frame
a_{iv}	Value of v^{th} dimension of feature vector $\mathbf{a_i}$
B	Number of bins
b_v	The bin along the v^{th} dimension
$\mathbf{C}$	The set of cliques in a graph
$\mathbf{C}_i$	The i^{th} clique in a graph
C	SVM's cost parameter
D	Number of dimensions in the video key frame feature vector
E	The set of bipartite edges across the visual and textual nodes
$\mathbf{F}$	A set of video key frames
$\mathbb{F}_i$	A set of frames associated with mention m_i
f_i	i^{th} video key frame
$\dot{f}_s$	The factor associated with a set of variables $\mathbf{X_s}$
G	A graph
K, K	A key entity cluster

$\mathbf{M}$	A set of textual mentions
m_i	Textual mention at position i
$\mathcal{N}$	Normal distribution
$\mathbf{N}$	A set of classes, animal names
$\mathbb{N}_i$	The set of names associated with frame f_i
n_l	Name at position l
n	A class label
Nb_i	The neighbors of clique $\mathbf{C}_i$
$\mathbf{P}$	A set of frame-mention pairs
R, R	A response entity cluster
$\mathbf{S}_{i,j}$	The sepset between cliques $\mathbf{C}_i$ and $\mathbf{C}_j$
$\mathbf{T}$	A set of textual nodes
t	A textual node comprising a mention and a name
$\mathbf{V}$	A set of visual nodes
v	A visual node comprising a frame and a name
$\mathbf{w_l}$	The set of weights corresponding to name n_l
X, X_i, Y	Stochastic variables
$\mathbf{X}_{\setminus i,j}$	The set $\mathbf{X}$ of all variables with X_i and X_j removed
x	Outcome of a random variable
y_l	Binary output label for name n_l
$\overline{y_l}$	Negation of y_l
Z	Normalization constant
Z_l	Normalization constant for the name n_l

Contents

List of Figures

List of Tables

Chapter 1

Introduction

It is estimated that it would take more than 5 million years to watch all the video that will cross global Internet Protocol (IP) networks each month by 2020[1]. Given the immensity of the video data around us, it is just insurmountable for human eyes to sift through all these videos, comprehend the matter and annotate them with labels or tags that describe the content correctly and completely. Nevertheless, current technologies largely rely on manually annotated tags to retrieve a video snippet of interest from this enormous collection. It is, therefore, imperative that we have tools that decipher videos automatically and index them with appropriate tags to make them 'searchable'.

Towards this generic goal of automatically deciphering video content, we take the case of wildlife documentaries with subtitles, and focus on the *entities* present in them, namely, the animals. State-of-the-art methods for identifying entities in vision and language view this as an alignment problem. A classical example is the alignment of names and faces [5, 25, 70]. These methods involve two aspects: (i) the use of a face detector to localize the faces present in the visuals and (ii) identifying entities relevant to the names in the text, e.g., by using a named entity recognizer and a coreference resolver. While these prerequisites are easily carried out for the name-face alignment task, they are not straightforward for the task of recognizing animals. Firstly, current 'animal' or general-purpose detectors are still at a very nascent stage. Dusart et al. [23] have solved the problem of recognizing animals from wildlife documentaries using manually created bounding boxes. But acquiring these bounding boxes is laborious and cumbersome, limiting the value of the approach. Therefore, in this thesis, we

[1] http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html (accessed May 15, 2017).

address the problem of recognizing entities such as animals, without relying on bounding boxes, tackling the problem of *object recognition'in-the-wild'*[2].

Second, named entity recognizers and coreference resolvers are tailored mostly towards names of people, locations and organizations, in a text-only setting. This brings two important requirements: (i) to design more generic entity analysis tools, that can cater to a broader class of entities and (ii) to develop language analysis tools that can leverage multimodal content. In this thesis, we address these requirements.

In particular, given a video documentary with subtitles, we tackle the following questions: a) What/ who are in the video key frames? (*Object recognition*) and b) What do the textual entity mentions refer to? (*Entity linking*). Next, we describe the challenges associated with these tasks.

1.1 Automatic Wildlife Recognition: Setup and Challenges

This nature documentary setup is both challenging and appealing from several perspectives.

- **On the vision side:** There are several challenges due to 1) the content, 2) our setup and 3) the rendering of the video. We elaborate on these below:

    1. *The content*: One of the key factors influencing the ease of recognition is the content of the videos. Animals are among the most difficult objects to recognize in images and videos, mainly due to their deformable bodies that often self occlude and the large variation they pose in appearance and depiction [6, 1]. Additionally, in the natural habitat, there are challenges due to camouflage and occlusion due to environment (sand, forest, water, snow etc.). Figure 1.1 illustrates these challenges on video key frames from our dataset. Compare these to scenarios such as those in Figure 1.2 that deals with much easier objects, or with Figure 1.3 that deals with animals, but in a much cleaner setting - where images are obtained from a Web search, and manually processed to ensure that the target animal is in a prominent view.

[2]Object recognition 'in-the-wild' refers to the fact that object recognition is done without localizing the objects of interest.

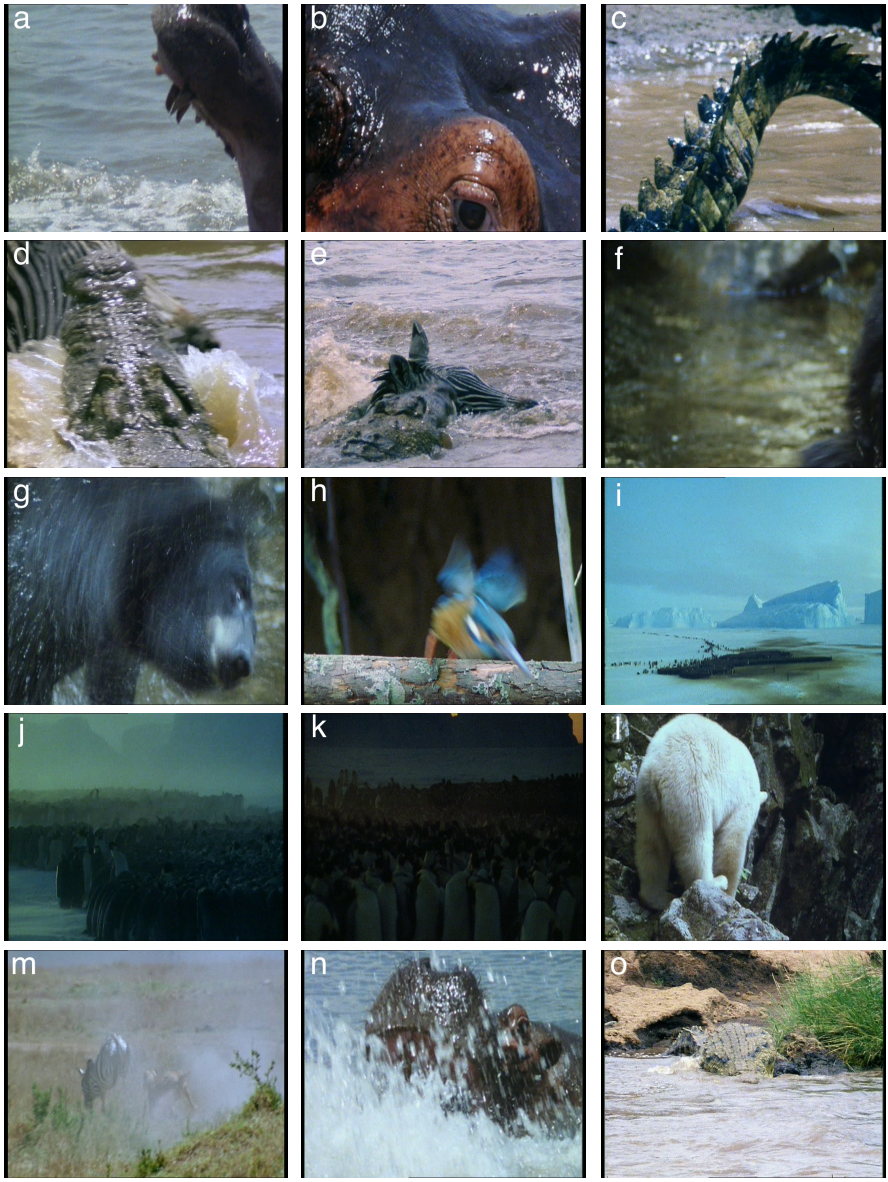


Figure 1.1: Challenges from the vision side: Random poses where key distinguishing features of the animal are absent (a,b,c), Multiple species in same image (d,e), Blurry images (f), The animal to be recognized is blurred out (g,h), The animal to be recognized is too far from the camera (i), Poor illumination (j,k), Self occlusion or auto-occlusion (l), Occlusion due to environment/ context and camouflage (m,n,o). All pictures above are from our wildlife dataset. ©BBC World

Figure 1.2: Images from the RGB-D object dataset. [51]

2. *Our setup*: While earlier work [71, 50] has focused on recognition of objects or people, after a careful selection of the regions of interest, we expose ourselves to the raw, unrefined data. We work in a setting where there are no visual demarcators, such as bounding boxes (see Figure 1.4). The absence of bounding boxes is challenging due to the following reasons. Firstly, the presence of an animal is not known - there could be several frames without animals. Second, if the frame has animals, there could be multiple animals of possibly different species. Third, this means the background can negatively influence the recognition. For example, a zebra on grasslands may now be very different from one in water.

   This *in-the-wild* setting is a more generic case and addresses a broader class of problems compared to that with bounding boxes. While this certainly entails more challenges, it is also more interesting - this means that the methods applied on this dataset are not restricted to curated images with neatly annotated bounding boxes, but can be applied to scenarios where bounding boxes are not available. It is liberating to discover that *we do not have to rely on object detectors for reliable object recognition.*

3. *The rendering*: As regards the rendering of the video, this is an *interlaced* video. *Interlaced scan*, in contrast to the other alternative namely *progressive scan*, proceeds in two passes - The first pass

Figure 1.3: Images from the "Animals with Attributes" dataset. [52]

Figure 1.4: Examples of a key frame with and without bounding box. Image on the right shows a bounding box in red indicating the presence of the white bear. ©BBC World

displays all odd numbered lines, from the top left corner to the bottom right corner. The second pass displays all the even numbered lines, filling in the gaps in the first scan. A field is an image that

Figure 1.5: Example of motion blur in a frame from an interlaced video. Figure courtesy: https://en.wikipedia.org/wiki/Interlaced_video (accessed May 15, 2017).

contains only half of the lines needed to make a complete picture. While the eye perceives the two fields as a continuous image due to persistence of vision, for automatic processing, these videos are more challenging. Because each interlaced video frame consists of two fields captured at different moments in time, interlaced video frames can exhibit *motion artifacts* known as *interlacing effects* or *combing*, if recorded objects move fast enough to be in different positions when each individual field is captured. These artifacts are more visible in still frames. Figure 1.5 shows an example of a frame from an interlaced video. Note that the blur due to motion is quite prominent. This makes the recognition far more challenging.

- **On the language side:** We have the subtitles[3], which are basically the written form of the spoken narration. It is generally agreed that written language is structurally elaborate, complex and formal, while spoken language is context-dependent and structurally simple [8]. In fact, in spoken language, there might be repetitions, incomplete sentences and interruptions. The subtitle excerpt below shows an example.

  But if you follow them for any length of time in their true home,
  these forests in West Africa,
  you discover that they are hunters.
  What's more, they hunt in teams
  and have a more complex strategy than any other hunting animal
  except...

[3]Subtitles and captions are often used interchangeably, although there is a subtle difference: captions include sound information such as 'laughter', or 'machinery starting up', while subtitles typically display only what is spoken by a character. Here, we do not make the distinction.

> (SCREECHING)
> ..except, of course, man.

Moreover, in the subtitles, there are no paragraph breaks. As a result, the end of a topic and the beginning of the next are not clear. For example, consider the subtitle snippet below.

> The splash tetra must have the most labour-intensive childcare of any fish.
> But his eggs are safer from predators on leaves rather than in the river.
> After two days of hard splashing, the fry emerge.
> Within minutes, this nervous herd will fragment into hundreds of individual families, as each stallion attempts to shepherd his mares and foals across.

In this example, the first three sentences are about the splash tetra, while the last is about the zebra. In spoken text such as subtitles, there are no markers to indicate the change of topic.

While we have tools such as [21, 54] that can process natural language text and resolve textual mentions, these systems are trained on well-written documents (e.g., news articles), and do not transfer well to subtitles.

These tools can also be used to detect entity mentions in the text. However, not all of them are pertinent to animals. Even when the mentions refer to animals, they are often so ambiguous (e.g. *'targets for the crocodile'* and *'the predators'*) that it is impossible to resolve them correctly without a holistic understanding of the context.

- **On the association of vision and language:** In video documentaries, vision and text in subtitles are not parallel, but complementary. The subtitles usually correspond to the voice over, and are meant to provide additional information to the viewer. They do not serve as a replacement for the visuals. This is in contrast to transcripts (often in the form of the so-called *video descriptions* or *screenplays*), that include audio-narrated descriptions of a video program's key visual elements, inserted into natural pauses in the program's dialogue. These typically provide a complete, precise description of the visuals, and are meant to make the video more accessible to visually impaired individuals. These are, however, rare and expensive to acquire compared to the subtitles or captions that are more often present. (See Figure 1.6 for a comparison between subtitles and transcripts.)

  This means that in contrast to most work on integrated vision and language in the literature, where textual descriptions are tightly linked to the image content, here the subtitles only serve as a weak supervisory signal.

00:17:08.8 --> 00:17:23.6: It's exhausting, especially if there's little to drink, 00:17:27.5 --> 00:17:27.7: so the legions march mostly while it's cool in the early morning and evening. 00:17:33.1 --> 00:17:34.7: For them, this migration is an essential journey. 00:17:38.2 --> 00:17:38.4: Unable to raise their young on land, 00:17:40.7 --> 00:17:40.9: these crabs must trek several kilometres each year to their ancestral home, the sea. 00:17:47.7 --> 00:18:42.0: The journey is perilous, for the exodus from the forest along traditional routes 00:18:47.2 --> 00:18:47.4: brings many of them into an alien world.	IN BLACK: We hear the faint sound of CAR TIRES running over the CONCRETE SEAMS of a highway. Eventually,... MAN'S VOICE (V.O.) Duncan, are you asleep? CUT TO: 1 INT. STATION WAGON - DAY 1 CLOSE ON DUNCAN, staring off, lost in thought. Some SUITCASES and COOLERS flank him. It's a little cramped. PULL BACK to reveal he's sitting in that ill-conceived back bench seat that faces out the rear of a vintage 1971 Buick Estate station wagon. MAN'S VOICE (O.S.) ... Duncan? CLOSE ON REARVIEW MIRROR. TRENT RAMSEY (MAN'S VOICE) glances back at Duncan. TRENT Duncan, are you sleeping?

Figure 1.6: Captions from our video on the left and screenplay/transcript (excerpt from the film 'The way way back') on the right: Note that the captions consist of the narration and the time pointers while the screenplay serves as a description of the video.

> Furthermore, the absence of bounding boxes in the visual data coupled with the presence of ambiguous mentions in text makes it harder to reliably tie together the entities in vision and language, that is, there are no ready examples to show the association. This problem is far more pronounced in a limited dataset with a large diversity.

Before describing our approaches to solve these problems, we turn to the all time gold standard of such tasks - the human brain.

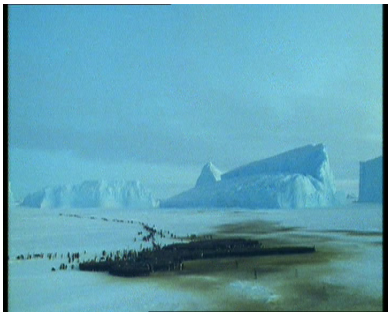
1.2 Inspiration: Vision, Language and the Human Brain

Vision and language are two faculties that have had remarkable impact on human lives. The origin of vision in the universe dates back to 543 million years ago, a period called the Cambrian period. This geological period saw a burst of apparently rapid evolution, dubbed the 'Cambrian explosion'. One of the most convincing hypotheses for causes of this diversification, is the *Light Switch* theory of Andrew Parker. In his book *'In the blink of an eye: how vision kick-started the big bang of evolution'* [68], Parker states that the evolution of eyes initiated an arms race that led to a rapid spate of evolution. Earlier than this, organisms may have had use for light sensitivity, but not for fast locomotion and navigation by vision.

The Merriam-Webster dictionary defines *vision* as '(i) the ability to see : sight or eyesight, (ii) something that you imagine : a picture that you see in your mind, (iii) something that you see or dream especially as part of a religious or supernatural experience'. Given the role that vision has played in shaping the universe and human lives, it is no surprise that this word has a connotation that goes beyond 'eyesight'.

Moving on to *language*, the Merriam-Webster dictionary defines *language* as '(i) the system of words or signs that people use to express thoughts and feelings to each other, (ii) any one of the systems of human language that are used and understood by a particular group of people, (iii) words of a particular kind'. Language is said to play a major role in allowing us to communicate complex concepts and harness our innate ability to form lasting social bonds with one another, distinguishing humans from the rest of the animal kingdom.

While vision and language are both potent, what is even more powerful is the integration of the two. Neuroscientists believe that, in general, interactions between multiple sensory systems (such as vision and audition) is beneficial [11, 31, 62] for at least two reasons. Firstly, each sensory system can provide 'missing pieces'. Second, when the two senses provide information about exactly the same object or event, combining the signals from each modality can enhance the accuracy of the resulting percept. While these hold for all sensory systems in general, vision and language are a particularly exciting combination. Vision and language are the two primary systems available for studying human perception and cognition, including those 'central' processes that are involved in all cognitive domains, such as attention, memory, and learning. Moreover, in the human brain, the two systems often operate in concert, for example, when we discuss aspects of the world around us [30].



With temperatures of 70 below, and in terrible storms, the penguins huddle tightly together for warmth.

The hippopotamus. Supported by the water, they use less energy than they would on land.

Figure 1.7: *Identifying objects on the vision side can be largely aided by the associated text.* The figure shows two frames together with corresponding subtitles from the video. On the left is a key frame showing a penguin huddle, shot from a distance. On the right is a frame covering parts of two hippos. In either case, it is difficult to identify what is in the frame without the supporting text. ©BBC World

One of the most common examples of vision-language combination, that currently occupies over 80% of the cyber space is *video*. It is interesting to understand how we process videos. Human viewers watching the videos and listening to commentary effortlessly gain a holistic understanding of the story. In fact, we can even deal with unfamiliar content on the visual or linguistic side. For example, consider this passage:

> What bird has the most elaborate, the most complex, the most beautiful song in the world? There are lots of contenders, but this bird must be one of them the superb lyrebird of southern Australia. (SHRILL CHIRPING) He clears a space in the forest to serve as his concert platform.

A person unfamiliar with the lyrebird can quickly grasp what a lyrebird looks like by watching the video clipping associated with this text. Likewise, one can also learn the name of a visually familiar object, from the text that occurs with the visuals.

This *complementarity* aspect inherent to video documentaries with subtitles corresponding to the voice over, is further illustrated in Figures 1.7 and 1.8. Figure 1.7 shows how the text aids recognition of animals in pictures. Using the text, subjects that are out of focus, or too far to be recognized or images that miss key distinguishing characteristics of an animal can be recognized with ease.

In the Dadia Mountains in Greece, golden eagles hunt over open forest, but their long wings create problems when chasing their prey among the trees. They have found an unlikely alternative prey. Soaring, as a search technique, is equally effective for finding tortoises. [...] The armoured shell presents an intriguing challenge. It simply doesn't have the right tools for the job. Its solution is ingenious. It flies to a favoured site, where a rocky outcrop acts like a natural anvil. It then gains height. It then safely parachutes into the clearing. The impact separates the tortoise shell into two halves, like loosening a lid.

Figure 1.8: *Understanding language can be facilitated by vision.* The above figure shows an excerpt of the subtitle text together with the associated video key frames. The visuals enable us to understand the sequence of events: a flying eagle spots the tortoise, it picks the tortoise using the claws and drops it from a height. ©BBC World

Likewise, the visuals also simplify understanding of the text as demonstrated in Figure 1.8. In some cases, it is even impossible to understand the text without the use of the corresponding visuals. Figure 1.9 shows an example. Here she_1 refers to the kingfisher and she_2 refers to the mink. While there is no ambiguity in these two cases, resolving she_3 is not straightforward. This piece of text might suggest that she_3 refers to the kingfisher, but in reality, it refers to the mink. The use of the associated frames makes this clear. For a viewer, watching the visuals in addition to listening to the narration gives a complete account of the story portrayed in the video.

Inspired by this, we explore methods that can harness this synergy between the visual and textual modalities to better comprehend videos for better multimedia indexing. Next, we outline the motivations and goals of this research.

[...] this daybreak finds the kingfisher still digging. She$_1$ must be desperate. [...] A mink. I thought it was an otter when it burst out from the bank. One kingfisher had dived to safety, but which one? It was impossible to tell. The mink had been waiting in ambush, hidden, even from me, almost certainly attracted by the kingfishers' frantic whistling. She$_2$ stashed the first bird and returned, sure that there was another. But one kingfisher got lucky. She$_3$ spotted me.

Figure 1.9: An example of a subtitle excerpt together with the associated frames. ©BBC World

1.3 Motivations and Goals

We formulate the following research questions which have mainly driven the research conducted in this thesis.

- Can we build object recognition models that can deal with a noisy, *'in-the-wild'* setting, as opposed to using clean, curated data, with carefully annotated labels? Can these models work with subtitles that are complementary to the vision, in lieu of transcripts or textual descriptions that are more parallel and provide a complete, accurate account of what is shown in the images or the video? Can we leverage external labeled datasets to learn object recognition models to overcome the lack of sufficient, reliable training data? How can these models be adapted to a multi-modal context involving vision and language?

- Can we build image representations and object recognition models that deal with the challenges above (noisy, *'in-the-wild'* setting, and complementary data), while also coping with the lack of external training data? How well do these models transfer to unseen images of an entirely different domain, captured across diverse topographical regions under vastly varying environmental conditions and illumination settings?

- In a multi-modal setting such as videos with subtitles, is there a model/representation that can encode dependencies within and across the modalities? Can we capitalize on the inherent characteristics of video documentaries (such as temporal continuity and the said interdependence) to build models that can jointly recognize the content of the video key frames and resolve the textual mentions in the subtitles? Can we automatically detect textual mentions that are relevant for the context (e.g., mentions relevant to animals, or those relevant to electronic equipment)?

In pursuit of answers to these questions, we made a series of contributions that are briefed next.

1.4 Contributions and Thesis Outline

We start by targetting the first set of research questions in Section 1.3. We propose a weakly supervised approach to accurately associate animals in the video with their names in subtitles in order to assign tags or labels to video frames. Here, we propose a feature transformation that allows us to split an image into components, and represent the image in terms of presence or absence of components. Building on this feature transformation and leveraging external labeled datasets to cope with the lack of sufficient, reliable training data, we propose the first ever framework for *domain adaptation in a multi-modal context for multi-label classification*. The basic idea is that we start from classifiers trained on external data (the source, in our setting - ImageNet), and iteratively adapt them to the target dataset using textual cues from the subtitles. We show that by training classifiers on an external labeled dataset, and adapting them iteratively to the target dataset, using textual cues, the accuracy of classification can be improved by a significant margin - the accuracy of our approach is significantly better than a) a purely vision-based approach or b) purely text-based approach or c) an approach that uses both text and vision, but without labeled examples or d) an approach that uses both text and vision, and labeled (out-of-domain) examples, but without the adaptive learning. This work is presented in Chapter 3 and was published as:

Venkitasubramanian, A. N., Tuytelaars, T., and Moens, M.-F. Wildlife recognition in nature documentaries with weak supervision from subtitles and external data. Pattern Recognition Letters. Elsevier 81 (2016), 63–70.

Then, we target the second set of research questions. We study different image representations and recognition models with the aim of addressing the lack of external labeled training data, in addition to the challenges due to noisy, *'in-the-wild'* setting, and complementary data. In particular, we investigate a support vector machine on top of activations of a pre-trained convolutional neural network, as well as a naive Bayes framework on a *'bag-of-activations'*, where each element of the bag is considered separately. This *'bag-of-activations'* paradigm allows key components in the image to be isolated, in spite of vastly varying backgrounds and image clutter, without an object detection or image segmentation step. The methods are also evaluated based on how well they transfer to unseen camera-trap images captured across diverse topographical regions under different environmental conditions and illumination settings, involving a large domain shift. This work is covered in Chapter 4 and is based on:

Venkitasubramanian, A. N., Tuytelaars, T., and Moens, M.- F. Learning to Recognize Animals by Watching Documentaries: Using Subtitles as Weak Supervision. In Proceedings of the EACL Workshop on Vision and Language (2017).

Next, we address the final set of research questions. We capitalize on the inherent characteristics of video documentaries (such as temporal continuity and the interdependence within and across visual and textual modalities) to build models that can recognize the content of the video key frames, and resolve the textual mentions in the subtitles. Moving on from approaches that predict on a frame-by-frame basis, we use a structured predictor that jointly addresses the questions: a) What do the textual entity mentions refer to? and b) What/ who are in the video key frames? In Chapter 5, we propose a novel weakly supervised framework that jointly tackles *entity analysis tasks in vision and language*. We use a Markov Random Field (MRF) to incorporates beliefs using independent methods for the textual and visual entities. These beliefs are propagated across the modalities to jointly derive the entity labels. We apply the framework to a dataset of wildlife documentaries with subtitles and show that this integrated modeling yields significantly better performance over text-based and vision-based approaches. We show that textual mentions that cannot be resolved using text-only methods are resolved correctly using our method. This framework tackles several challenges: (i) the absence of visual demarcators, (ii) the lack of reliable training examples, (iii) the presence of irrelevant mentions, and relevant but ambiguous mentions on the text. The

work presented in this chapter is based on:

Venkitasubramanian, A. N., Tuytelaars, T., and Moens, M.- F. Entity linking across vision and language. Multimedia Tools and Applications (2017) DOI:10.1007/s11042-017-4732-8

The organization of the rest of this dissertation is as follows: Chapter 2 presents an overview of fundamental concepts that are essential for a deeper understanding of this thesis. Chapters 3, 4 and 5 are based on the author's published or accepted peer-reviewed papers or journal articles. Chapter 3 presents a multimodal domain adaptation framework for wildlife recognition. Chapter 4 studies image representations and object recognition models for wildlife recognition in the absence of external labeled training data. Chapter 5 jointly tackles the problems of object recognition on the vision side and entity linking on the text side. Finally, conclusions, contributions and future work perspectives based on the work conducted in this thesis are summarized in Chapter 6.

Chapter 2

Fundamentals

This chapter contains brief reviews of basic modeling concepts which serve as a fundamental theoretical background for the remainder of this text. All of this material focuses only on the concepts required for a deeper understanding of the upcoming parts of the thesis text. While these fundamentals have not been covered in their entirety, they are accompanied with references for further reading. The reader already familiar with these elementary concepts may safely skip the parts of this chapter discussing them.

A large body of the work presented in this thesis is at the intersection of three domains: a) Machine Learning, b) Computer Vision and c) Natural Language Processing, and falls into the broad field of Artificial Intelligence (AI).

Machine Learning: Machine learning is the branch of computer science that gives computers the ability to learn without being explicitly programmed. Machine learning explores the study and construction of algorithms that can learn from and make predictions on data rather than following strictly static program instructions. This allows algorithms to learn through experience, and do things we don't know how to make programs for. For problems such as understanding and interpreting vision and language, a *'supervised'* paradigm with neatly annotated input-label pairs that enable 'training' of the system, is used most often. Inspired by the ability of people to learn on the fly, this thesis explores *'weakly-supervised'* methods that do not rely on curated data, and can deal with a realistic setting instead. A remarkable development in this field is the renewed surge of interest in *Artificial Neural Networks* (ANN) with *deep* architectures, consisting of multiple layers of computational units called *neurons*. The so-called *deep neural networks* have been successfully used for various tasks. An interesting alternative to these vanilla deep neural networks is

the *convolutional neural network* that makes explicit assumptions about inputs, relevant for images. Throughout this thesis, we use visual features extracted from a *convolutional neural network*, pre-trained on a million images. Machines driven by AI technology are able to perform consistent, repetitious actions without human shortcomings, such as fatigue, emotion and limited time.

Computer Vision: Computer vision (often dubbed vision) is the science and technology of machines that see. It is concerned with the automatic extraction, analysis and understanding of useful information from a single image or a sequence of images. It involves the development of a theoretical and algorithmic basis to achieve automatic visual understanding. Computer vision is inspired by the human vision system which is the richest sense that we have. To us vision seems easy, but in reality we are processing around 60 images per second with millions of points (pixels) in each image. In fact, over half the human brain is involved in processing visual information, and this seems a good indication that this is a very complex task. Ultimately computer vision aims at emulating human vision but this is still a long way away. A large body of this thesis revolves around identification of objects such as animals shown in videos or images.

Natural Language Processing: Natural Language Processing (NLP) is a field of research between computer science and linguistics which aspires towards the automated analysis, representation, transformation and generation of natural language texts by means of computer algorithms. Interest in NLP began in earnest in 1950 when Alan Turing published his paper titled "Computing Machinery and Intelligence", from which the so-called Turing Test emerged. Turing asserted that a computer could be considered intelligent if it could carry on a conversation with a human being without the human realizing he/she were talking to a machine; such is the importance of comprehending language as a measure of intelligence. Towards the goal of understanding language, this thesis explores methods to automatically map textual mentions to real-world entities, in addition to the object recognition in vision.

Artificial Intelligence: It is technology and a branch of computer science that aims to develop 'intelligent' machines and software, that is, any device that perceives its environment and takes actions that maximize its chance of success at reaching some goal. The field has various applications in several diverse domains, ranging from *self-driving cars* that have logged over 300,000 accident-free miles to *cognitive prostheses* or brain implants that can perform the role of a part of the brain that has been damaged and *robot-scientists* such as "Adam"[1] that discover new scientific knowledge, coming up with their own hypotheses and testing them. In this thesis, we explore methods to comprehend

[1]https://en.wikipedia.org/wiki/Robot_Scientist (accessed May 15, 2017).

the visual and textual content of videos, with special focus on the entities shown and mentioned.

In this chapter, we therefore cover relevant concepts from these domains. This chapter is organized as follows: Section 2.1 presents a short overview of statistics and machine learning concepts that form the backbone of our work. Section 2.2 covers relevant concepts from the computer vision side, while Section 2.3 discusses the language side.

2.1 Statistics and Machine Learning

In this section, we cover statistics and machine learning concepts that will be used throughout this thesis. We start with an introduction of basic concepts of probability theory (Section 2.1.1), then move to describe Gaussian distributions (Section 2.1.2). In Section 2.1.3 we describe several probabilistic graphical models, together with inference algorithms. Finally, in Section 2.1.4 we present a brief outline of the Expectation-Maximization algorithm.

2.1.1 Basic Concepts of Probability Theory

Imagine an event happening with several different possible outcomes (e.g., a flip of a coin or a roll of a dice). We would like to know the probability that a coin will land heads. But, what is *probability*? A *frequentist* interpretation of probability represents probabilities as long run frequencies of events (e.g., if we flip a coin many times, we expect it to land heads about half of the times). On the other hand, the *Bayesian* interpretation of probability models uncertainty about events that do not have long term frequencies (e.g., we believe that the coin is equally likely to land tails or heads in the next flip). In short, in this interpretation, probability is used to quantify our *uncertainty* or our *belief* about something. However, regardless of the actual interpretation, the rules of probability theory remain the same.

In this section we provide a short introduction to the probability theory notions used throughout this dissertation. A more in-depth introduction can be found in [9, 67, 4].

- *Probability theory* is a mathematical theory to describe and analyze situations where randomness or uncertainty are present. Any such situation will be referred to as a *random experiment*.

- The *sample space*, denoted S is the set of all possible outcomes in a random experiment. The sample space associated with the flip of a coin is $S = \{H, T\}$, while that associated with the rolling of a dice is $S = \{1, 2, 3, 4, 5, 6\}$.

- An *event* X is a set of outcomes of an experiment, that is, a subset of the sample space ($X \subseteq S$). The occurrence of a head when tossing a coin is an event denoted by $X = \{H\}$. The occurrence of an even number when rolling a dice is an event denoted by $X = \{2, 4, 6\}$.

- Probability is a mapping $p : X \rightarrow [0, 1]$ from events $X \in S$ to real values. It satisfies the following conditions:

  1. $0 \leq p(X) \leq 1; \forall X \in S$
  2. $p(S) = 1$
  3. Any countable sequence of *pairwise disjoint* events $X_1, X_2, \ldots$ (*i.e.*, $X_i \cap X_j = \varnothing$ whenever $i \neq j$) satisfies

$$p\left(\bigcup_{i=1}^{\infty} X_i\right) = \sum_{i=1}^{\infty} p(X_i)$$

  An important corollary of these axioms is that $p(\overline{X}) = 1 - p(X)$; where $\overline{X} = S - X$.

- *Joint probability* is the combined probability of multiple variables defined over the same space S. For two random variables X and Y, we denote their joint probability as $p(X, Y)$. This is actually the joint distribution over all possible outcomes for X and Y happening together. The probability $p(X, Y)$ is computed as $p(X \cap Y)$. The probability of a union of two events X and Y is computed as $p(X \cup Y) = p(X) + p(Y) - p(X \cap Y)$. In case when two events do not overlap (i.e., they are disjoint), that probability of at least one of them happening becomes $p(X \cup Y) = p(X) + p(Y)$.

- *Conditional probability and independence*: If Y is an event with non-zero probability, the *conditional probability* of any event X given Y denoted by $p(X|Y)$ is defined as

$$p(X|Y) = \frac{p(X \cap Y)}{p(Y)}$$

  In other words, $p(X|Y)$ is the probability of the event X *after observing the occurrence* of the event Y. The conditional probability is also called the *posterior probability* since it is computed after the event Y has been observed. Two events X and Y are said to be *independent* if and only if

$p(X \cap Y) = p(X) * p(Y)$, or equivalently $p(X|Y) = p(X)$. The condition of independence is equivalent to saying that observing Y does not have any effect on the probability of X. If X and Y are independent, it consequently holds $p(X, Y) = p(X \cap Y) = p(X|Y) * p(Y) = p(X) * p(Y)$.

- The *product rule*: Using the above expression, the joint probability $p(X, Y)$ can be obtained using the conditional probability $p(X|Y)$ and $p(Y)$:

$$p(X, Y) = p(X|Y) * p(Y)$$

- *Marginal probability* and the *sum rule*: Calculating the probability of only one variable in a joint probability is through a process called *marginalization*. Given a joint distribution of two events $p(X, Y)$, the marginal probability $p(X)$ is computed by summing the probability over all possible states of Y:

$$p(X) = \sum_{Y} p(X|Y) * p(Y)$$

  Here, $p(X)$ is often called the *prior* or *prior distribution*, as it reflects the probability of the event X in advance, that is, before anything is known about the outcome of the event Y.

These basic probability concepts are used throughout this thesis (in Chapters 3, 4 and 5).

2.1.2 Gaussian Distribution

We introduce here one of the most important probability distributions for continuous variables, called the *normal* or *Gaussian* distribution. For the case of a single real-valued variable x, the Gaussian distribution is defined as

$$\mathcal{N}(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(x-\mu)^2/2\sigma^2}$$

This distribution is governed by two parameters: μ called the *mean*, and σ^2, called the *variance*. The square root of the variance, given by σ, is called the *standard deviation*, and the reciprocal of the variance, written as $1/\sigma^2$, is called the *precision*.

From the above equation, we see that the Gaussian distribution satisfies

$$\mathcal{N}(x|\mu, \sigma) > 0$$

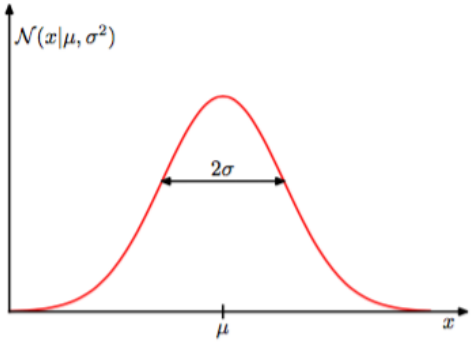


Figure 2.1: Plot of the univariate Gaussian showing the mean μ and the standard deviation σ.
Figure courtesy: [9]

Also it is straightforward to show that the Gaussian is normalized, so that

$$\int_{\infty}^{\infty} \mathcal{N}(x|\mu, \sigma)dx = 1$$

Figure 2.1 shows a plot of the Gaussian distribution. We revisit this distribution in Chapter 4.

2.1.3 Probabilistic Graphical Models

A *graph* comprises *nodes* (also called *vertices*) connected by *links* (also known as *edges* or *arcs*). In a *probabilistic graphical model*, each node represents a random variable (or group of random variables), and the links express probabilistic relationships between these variables. The graph then captures the way in which the joint distribution over all of the random variables can be decomposed into a product of factors each depending only on a subset of the variables.

We start with a brief introduction of a *bipartite graph*, and then move on to describe two major classes of probabilistic graphical models. The first class of models is *Bayesian networks*, also known as *directed graphical models*, in which the links of the graphs have a particular directionality indicated by arrows. The other major class of graphical models are *Markov random fields*, also known as *undirected graphical models*, in which the links do not carry arrows and have no directional significance. Directed graphs are useful for expressing causal relationships between random variables, whereas undirected graphs are better suited to express soft constraints between random variables [9]. What follows is

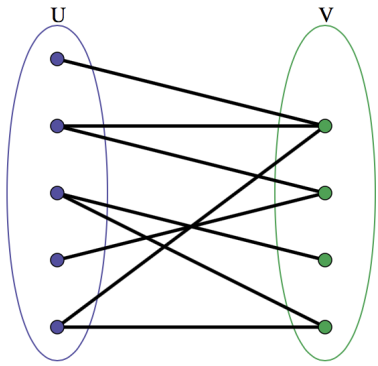


Figure 2.2: Example of a bipartite graph. Blue dots denote one vertex set (part of the graph), while the green dots represent another vertex set.
Figure courtesy: https://en.wikipedia.org/wiki/Bipartite_graph (accessed May 15, 2017).

a brief outline of these concepts. For a more detailed overview of probabilistic graphical models, we refer the interested reader to [49, 9].

Bipartite graph

A bipartite graph (or bigraph) is a graph whose vertices can be divided into two disjoint sets U and V, such that every edge connects a vertex in U to one in V. A bipaprtite graph is denoted as $G = (U, V, E)$, where U and V are the vertex sets (usually called the *parts* of the graph), and E is the set of edges. Figure 2.2 shows an example of a bipartite graph.

The concept of bipartite graph defined here is used later in this chapter in the context of a data structure called *factor graph*, and then in Chapter 5, where we build a bipartite graph consisting of textual and visual nodes, with edges going from one modality to another.

Bayesian Networks

A Bayesian network is a directed acyclic graph that represents a joint probability distribution (JPD) over a set of random variables. The network is defined by a pair $< G, \Theta >$, where G is the directed acyclic graph whose nodes $X_1, X_2, \ldots, X_D$ represent random variables, and edges represent the direct dependencies between these variables. The graph G encodes independence

assumptions, by which each variable X_i is independent of its non-descendents given its parents in G. The second component, Θ, denotes the set of parameters of the network. This set contains the parameter $\theta_{x_i|\pi_i} = p(x_i|\pi_i)$ for each realization x_i of X_i conditioned on π_i, the set of parents of X_i in G. Accordingly, the Bayesian network defines a unique JPD, namely:

$$p(X_1, X_2, \ldots, X_D) = \prod_{i=1}^{D} p(X_i|\pi_i) = \prod_{i=1}^{D} \theta_{X_i|\pi_i}$$

Generative Models

There are many situations in which we wish to draw samples from a given probability distribution. One technique particularly relevant to graphical models is the *ancestral sampling*. Corresponding to a directed acyclic graph, we consider a joint distribution $p(x_1, \ldots, x_D)$ over D variables that factorizes as follows:

$$p(X_1, X_2, \ldots, X_D) = \prod_{i=1}^{D} p(X_i|\pi_i)$$

where π_i denotes the parents of X_i. We suppose that the variables have been ordered such that there are no links from any node to any lower numbered node, in other words each node has a higher number than any of its parents. Our goal is to draw a sample $X_1, X_2, \ldots, X_D$ from the joint distribution. To do this, we start with the lowest-numbered node and draw a sample from the distribution $p(X_1)$, which we call x_1. We then work through each of the nodes in order, so that for node k we draw a sample from the conditional distribution $p(X_k = x_k|\pi_k)$ in which the parent variables have been set to their sampled values. Note that at each stage, these parent values will always be available because they correspond to lower-numbered nodes that have already been sampled. Once we have sampled from the final variable X_D, we will have achieved our objective of obtaining a sample from the *joint distribution*. To obtain a sample from some *marginal distribution* corresponding to a subset of the variables, we simply take the sampled values for the required nodes and ignore the sampled values for the remaining nodes. For example, to draw a sample from the distribution $p(X_2, X_4)$, we simply sample from the full joint distribution and then retain the values x_2, x_4 and discard the remaining values $\{x_j \neq 2, 4\}$.

For practical applications of probabilistic models, it will typically be the higher-numbered variables corresponding to terminal nodes of the graph that represent the *observations*, with lower-numbered nodes corresponding to *latent variables*. The primary role of the latent variables is to allow a complicated distribution

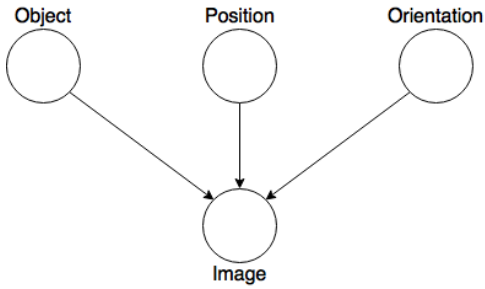


Figure 2.3: A graphical model representing the process by which images of objects are created, in which the identity of an object (a discrete variable) and the position and orientation of that object (continuous variables) have independent prior probabilities. The image (a vector of pixel intensities) has a probability distribution that is dependent on the identity of the object as well as on its position and orientation.
Figure courtesy: [9]

over the observed variables to be represented in terms of a model constructed from simpler conditional distributions.

We can interpret such models as expressing the processes by which the observed data arose. For instance, consider an object recognition task in which each observed data point corresponds to an image (comprising a vector of pixel intensities) of an object. In this case, the latent variables might have an interpretation as the position and orientation of the object. Given a particular observed image, our goal is to find the posterior distribution over objects, in which we integrate over all possible positions and orientations. We can represent this problem using a graphical model of the form show in Figure 2.3.

The graphical model captures the causal process [69] by which the observed data was generated. For this reason, such models are often called *generative models*.

Naive Bayes

An interesting graphical structure that is often used with classification is the *naive Bayes model*. In this model, we use conditional independence assumptions to simplify the model structure. Suppose our observed variable consists of a D-dimensional vector $x = (x_1, \ldots, x_D)^T$, and we wish to assign observed values of x to one of N classes. This observed variable could be a text document represented as a *Bag of Words* (BoW) or an image denoted as a *histogram or*

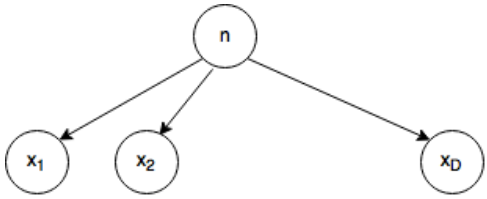


Figure 2.4: A graphical representation of the 'naive Bayes' model for classification. Conditioned on the class label n, the components of the observed vector $x = (x_1, \ldots, x_D)^T$ are assumed to be independent.

Bag of Visual Words (BoVW), for instance. The key assumption of the naive Bayes model is that, conditioned on the class, the distributions of the input variables $(x_1, \ldots, x_D)$ are independent. The graphical representation of this model is shown in Figure 2.4.

The classification is outlined below. We are interested in inferring the class n associated with the input x. First, the input x (which can be a document or image) is represented as a set of words or features.

$$p(n|x) = p(n|x_1, \ldots, x_D)$$

Using Bayes' rule,

$$p(n|x_1, \ldots, x_D) = \frac{p(x_1, \ldots, x_D|n) * p(n)}{p(x_1, \ldots, x_D)}$$

Here, $p(n)$ is called the *prior* and it allows incorporation of *prior knowledge* about the class distribution. As far as the prediction of class label is concerned, the term $p(x_1, \ldots, x_D)$ is a constant (over different values of n). So, this term can be ignored.

$$p(n|x_1, \ldots, x_D) \propto p(x_1, \ldots, x_D|n) * p(n)$$

Next, we employ the *naive Bayes assumption* which states that the features are conditionally independent of each other given the class labels.

$$\begin{aligned} p(x_1, \ldots, x_D|n) &= p(x_1|n) * p(x_2|n) * \ldots * p(x_D|n) \\ &= \prod_{i=1}^{D} p(x_i|n) \end{aligned}$$

Combining the above equations, we have

$$p(n|x) \propto \prod_{i=1}^{D} p(x_i|n) * p(n)$$

An important question is how will the probabilities $p(x_i|n)$ for the different features be computed. In this dissertation, we explore two approaches: a) Gaussian naive Bayes and b) Multinomial naive Bayes. These are briefly described below.

1. *Gaussian naive Bayes*: When dealing with continuous data, a typical assumption is that the continuous values associated with each class are distributed according to a Gaussian distribution. For example, suppose the data contain a continuous attribute, X_i. We first segment the data by the class n, and then compute the mean and variance of X_i for each class. Let μ_n be the mean of the values in X_i associated with class n, and let σ_n^2 be the variance of the values in X_i associated with class n. Then, the probability $p(X_i = x_i|n)$ can be computed using the Normal distribution parameterized by μ_n and σ_n^2. That is,

$$p(X_i = x_i|n) = \frac{1}{\sigma_n\sqrt{2\pi}} e^{-(x_i-\mu_n)^2/2\sigma_n^2}$$

2. *Multinomial naive Bayes*: When the feature vectors denote frequencies or histograms, the probability $p(X_i = x_i|n)$ can be computed using a ratio of frequencies as follows:

$$p(X_i = x_i|n = \nu) = \frac{|(X_i = x_i) \cap (n = \nu)|}{|n = \nu|}$$

The concepts of generative model and naive Bayes are used in Chapters 3 and 4 in the context of classifying animals present in a video key frame.

Markov Random Field

A *Markov random field*, also known as a *Markov network* or an *undirected graphical model* [47], has a set of nodes each of which corresponds to a variable or group of variables, as well as a set of links each of which connects a pair of nodes. The links are undirected, that is they do not carry arrows.

These graphs have the property that any two nodes X_i and X_j that are not connected by a link must be conditionally independent given all other nodes in the graph. This conditional independence property can be expressed as

$$p(X_i, X_j|\mathbf{X}_{\setminus i,j}) = p(X_i|\mathbf{X}_{\setminus i,j}) * p(X_j|\mathbf{X}_{\setminus i,j})$$

where $\mathbf{X}_{\setminus i,j}$ denotes the set $\mathbf{X}$ of all variables with X_i and X_j removed. The factorization of the joint distribution must therefore be such that X_i and X_j

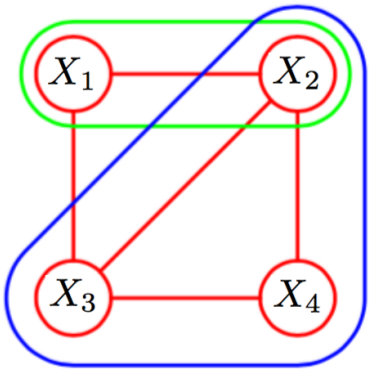


Figure 2.5: A four-node undirected graph showing a clique (outlined in green) and a maximal clique (outlined in blue).
Figure courtesy: [9]

do not appear in the same factor in order for the conditional independence property to hold for all possible distributions belonging to the graph.

This leads us to consider a graphical concept called a *clique*, which is defined as a subset of the nodes in a graph such that there exists a link between all pairs of nodes in the subset. In other words, the set of nodes in a clique is fully connected. Furthermore, a *maximal clique* is a clique such that it is not possible to include any other nodes from the graph in the set without it ceasing to be a clique. These concepts are illustrated by the undirected graph over four variables shown in Figure 2.5. This graph has five cliques of two nodes given by $\{X_1, X_2\}$, $\{X_2, X_3\}$, $\{X_3, X_4\}$, $\{X_4, X_2\}$, and $\{X_1, X_3\}$, as well as two maximal cliques given by $\{X_1, X_2, X_3\}$ and $\{X_2, X_3, X_4\}$. The set $\{X_1, X_2, X_3, X_4\}$ is not a clique because of the missing link from X_1 to X_4.

We can therefore define the factors in the decomposition of the joint distribution to be functions of the variables in the cliques. In fact, we can consider functions of the maximal cliques, without loss of generality, because other cliques must be subsets of maximal cliques. Thus, if $\{X_1, X_2, X_3\}$ is a maximal clique and we define an arbitrary function over this clique, then including another factor defined over a subset of these variables would be redundant. Let us denote the variables in a clique by $\mathbf{C}_i$. Then the joint distribution is written as a product of *potential functions* $\psi_i(\mathbf{C}_i)$ over the maximal cliques of the graph

$$p(\mathbf{x}) = \frac{1}{Z} \prod_{\mathbf{C}_i \in \mathbf{C}} \psi_i(\mathbf{C}_i)$$

Here the quantity Z, sometimes called the *partition function*, is a normalization constant and is given by

$$Z = \sum_{\mathbf{x}} \prod_{\mathbf{C}_i \in \mathbf{C}} \psi_i(\mathbf{C}_i)$$

which ensures that the distribution $p(\mathbf{x})$ is correctly normalized. By considering only potential functions which satisfy $\psi_i(\mathbf{C}_i) \geq 0$ we ensure that $p(\mathbf{x}) \geq 0$. Note that we do not restrict the choice of potential functions to those with a specific probabilistic interpretation as marginal or conditional distributions. This is in contrast to directed graphs in which each factor represents the conditional distribution of the corresponding variable, conditioned on the state of its parents.

Inference in Graphical Models

Inference is the mechanism used for answering queries using the distribution as our model of the world. In particular, this involves algorithms for computing the posterior probability of some variables given evidence on others. In what follows, we discuss a data structure called *factor graph*, that is often helpful for inference and two popular inference algorithms: the *sum-product* and the *max-product* algorithms.

Factor Graphs

Both directed and undirected graphs allow a global function of several variables to be expressed as a product of factors over subsets of those variables. *Factor graphs* make this decomposition explicit by introducing additional nodes for the factors themselves in addition to the nodes representing the variables. They also allow us to be more explicit about the details of the factorization.

Let us write the joint distribution over a set of variables in the form of a product of factors:

$$p(x) = \prod_{s} \dot{f}_s(\mathbf{x_s})$$

where $\mathbf{x_s}$ denotes a subset of the variables. For convenience, we denote the individual variables by X_i. Each factor $\dot{f}_s$ is a function of a corresponding set of variables $\mathbf{x_s}$.

In a factor graph, there is a node (depicted as usual by a circle) for every variable in the distribution, as was the case for directed and undirected graphs. There are also additional nodes (depicted by small squares) for each factor $\dot{f}_s(\mathbf{x_s})$ in the joint distribution. Finally, there are undirected links connecting each factor

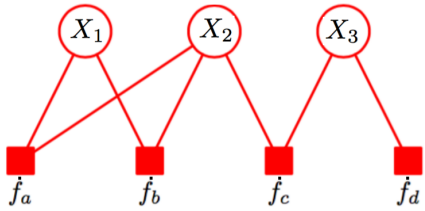


Figure 2.6: Example of a factor graph, which corresponds to the factorization $p(x) = \dot{f}_a(X_1, X_2)\dot{f}_b(X_1, X_2)\dot{f}_c(X_2, X_3)\dot{f}_d(X_3)$.
Figure courtesy: [9]

node to all of the variables nodes on which that factor depends. Consider, for example, a distribution that is expressed in terms of the factorization

$$p(x) = \dot{f}_a(X_1, X_2)\dot{f}_b(X_1, X_2)\dot{f}_c(X_2, X_3)\dot{f}_d(X_3).$$

This can be expressed by the factor graph shown in Figure 2.6. Note that there are two factors $\dot{f}_a(X_1, X_2)$ and $\dot{f}_b(X_1, X_2)$ that are defined over the same set of variables. In an undirected graph, the product of two such factors would simply be lumped together into the same clique potential. Similarly, $\dot{f}_c(X_2, X_3)$ and $\dot{f}_d(X_3)$ could be combined into a single potential over X_2 and X_3. The factor graph, however, keeps such factors explicit and so is able to convey more detailed information about the underlying factorization.

Factor graphs are said to be *bipartite* because they consist of two distinct kinds of nodes - the variable nodes and the factor nodes, and all links go between nodes of opposite type.

The Sum-Product Algorithm

With acyclic graphs, the sum-product algorithm allows efficient inference by exploiting the graph structure to achieve two things: (i) to obtain an efficient, exact inference algorithm for finding marginals; (ii) to allow computations to be shared efficiently, in situations where several marginals are required.

Consider the problem of finding the marginal $p(x)$ for particular variable node x. By definition, the marginal is obtained by summing the joint distribution over all variables except x. So, we have

$$p(x) = \sum_{\mathbf{x}\setminus x} p(\mathbf{x})$$

where $\mathbf{x} \setminus x$ denotes the set of variables in $\mathbf{x}$ with variable x omitted.

In a naive implementation, we would first evaluate the joint distribution and then perform the summations explicitly. We can, however, obtain a much more efficient algorithm by exploiting the conditional independence properties of the graphical model. The key insight that allows the effective computation of this expression is that the scope of the factors is limited, allowing us to "push in" some of the summations, performing them over the product of only a subset of factors.

The algorithm relies on the cliques in the graph, together with their associated *potentials*. The basic idea is that each node can send a *message* to each of its neighbors once it has received messages from all the other neighbors. That is, every clique $\mathbf{C}_i$ multiplies all incoming messages from its other neighbors with its initial clique potential. We refer the set of variables common to cliques $\mathbf{C}_i$ and $\mathbf{C}_j$ as the *sepset* between $\mathbf{C}_i$ and $\mathbf{C}_j$. $\mathbf{S}_{i,j} = \mathbf{C}_i \cap \mathbf{C}_i$. Now, to pass a message to clique $\mathbf{C}_j$, we sum out all variables except this sepset $\mathbf{S}_{i,j}$, and send the resulting factor to $\mathbf{C}_j$.

The message $\delta_{i \to j}$ from clique $\mathbf{C}_i$ to clique $\mathbf{C}_j$ is computed using the following *sum-product message passing* computation

$$\delta_{i \to j} = \sum_{\mathbf{C}_i - \mathbf{S}_{i,j}} \psi_i * \prod_{k \in (Nb_i - \{j\})} \delta_{k \to i}$$

where Nb_i refers to the neighbors of $\mathbf{C}_i$. The node's belief is updated by multiplying all incoming messages, together with the node's current belief. The belief β_j associated with clique $\mathbf{C}_j$ is obtained as follows:

$$\beta_j = \psi_j * \prod_{k \in (Nb_j)} \delta_{k \to j}$$

To illustrate this, consider the example of Figure 2.7, and assume the task is to compute *p(Job)*, denoted by $p(J)$. The set of maximal cliques associated with this network is depicted in Figure 2.8.

Our first step is to generate a set of initial potentials associated with the different cliques. The initial potential $\psi_i(\mathbf{C_i})$ is computed by multiplying the initial factors assigned to the clique $\mathbf{C_i}$). For example, $\psi_5(J, L, G, S) = p(L|G) * p(J|L, S)$.

We can then start with the first clique $\mathbf{C_1}$. We can eliminate C by performing $\sum_C \psi_1(C, D)$. We send it as a "message" $\delta_{1 \to 2}(D)$ to $\mathbf{C_2}$. Then, in $\mathbf{C_2}$, we define $\beta_2(G, I, D) = \delta_{1 \to 2}(D) * \psi_2(G, I, D)$. We then eliminate D to get a factor over G, I. The resulting factor is $\delta_{2 \to 3}(G, I)$, which is sent to $\mathbf{C_3}$. In $\mathbf{C_3}$: We

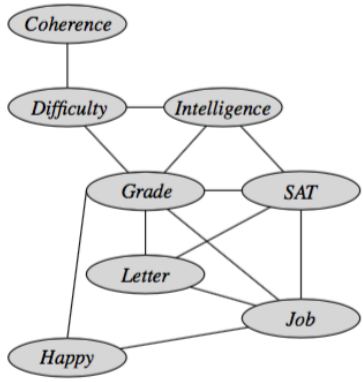


Figure 2.7: Example of a Markov network.Figure courtesy: [49]

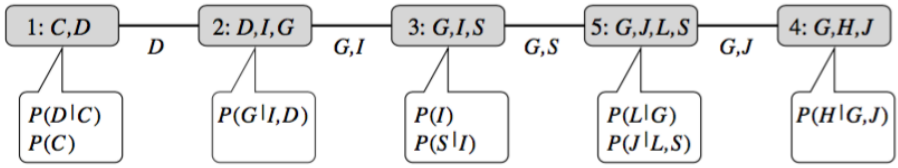


Figure 2.8: Set of maximal cliques associated with Figure 2.7. The random variables of Figure 2.7 are referred here by their initials (C for Coherence, D for Difficulty etc.). Gray rectangles on the top denote the cliques, while the white boxes below them represent the probabilities associated.
Figure courtesy: [49]

define $\beta_3(G,S,I) = \delta_{2\to3}(G,I) * \psi_3(G,S,I)$ and eliminate I to get a factor over G,S, which is $\delta_{3\to5}(G,S)$. Then, in $\mathbf{C_4}$, we eliminate H by performing $\sum_H \psi_4(H,G,J)$ and send out the resulting factor as $\delta_{4\to5}(G,J)$ to $\mathbf{C_5}$. Finally, in $\mathbf{C_5}$, we define $\beta_5(G,J,S,L) = \delta_{3\to5}(G,S)*\delta_{4\to5}(G,J)*\psi_5(G,J,S,L)$. Figure 2.9 illustrates this.

When applied to graphs with loops or cycles this algorithm is known as *loopy belief propagation* [32].

The Max-Product algorithm

The sum-product algorithm allows us to take a joint distribution $p(\mathbf{x})$ expressed as a factor graph and efficiently find marginals over the component variables.

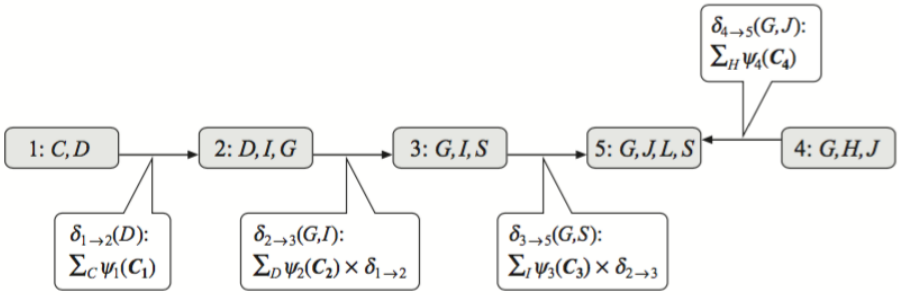


Figure 2.9: Illustration of the sum-product algorithm.Figure courtesy: [49]

Two other common tasks are (i) to find a setting of the variables that has the largest probability and (ii) to find the value of that probability. These can be addressed through a closely related algorithm called *max-product* algorithm. The max-product algorithm is identical to the sum-product algorithm except that summations are replaced by maximizations [9]. A special case of this algorithm is the *Viterbi algorithm*, used for finding the most likely sequence of hidden states in Markov chains and hidden Markov models.

The Markov random field and the inference algorithms summarized here form the bulk of Chapter 5, and are used for recognizing entities both in text and vision.

2.1.4 Expectation-Maximization

An elegant and powerful method for finding maximum likelihood solutions for models with latent variables is called the expectation-maximization algorithm, or EM algorithm [18, 61].

EM is a deterministic iterative algorithm well suited for dealing with incomplete data. It alternates between: (1) performing an *expectation step* (*E-step*), where a function for the expectation of the (log)-likelihood evaluated using the current estimate for the parameters is created (i.e., posterior probabilities are computed for the latent variables based on the current estimates of the parameters), and (2) a *maximization step* (*M-step*), in which parameters are computed by maximizing the expected (log)-likelihood found during the previous E-step (i.e., the parameters are re-estimated in order to maximize the likelihood function). Subsequently, these parameter estimates are again employed to determine the

distribution of the latent variables in the next E-step and the process converges to a (local) optimum.

The EM algorithm is used in Chapter 3, for learning from a source dataset and iteratively adapting to the target dataset.

2.2 Computer Vision

In this section, we discuss computer vision concepts that have been used throughout the thesis text. We start this section with brief introductions of *object detection* and *object recognition*. We then move on to a discussion on Convolutional Neural Networks (Section 2.2.1). For a more thorough treatment of these topics we refer the reader to [95, 85].

Object detection is the technology that deals with detecting instances of semantic objects of a certain class (such as humans, buildings, or cars) in digital images and videos. Currently, we have *face detectors* to localize the faces present in a picture (see Figure 2.10). These are some of the most successful examples of object detection. In fact, such algorithms are built into most of today's digital cameras to enhance auto-focus and into video conferencing systems to control pan-tilt heads. Additionally, there are also special-purpose detectors to detect pedestrians and vehicles (Figure 2.11 shows an example). Such detectors can be used in automotive safety applications, e.g., detecting pedestrians and cars from moving vehicles [55]. For the challenging task of wildlife recognition, there are no reliable detectors yet.

General object recognition falls into two broad categories, namely *instance recognition* and *class recognition*. The former involves recognizing a known 2D or 3D rigid object, potentially being viewed from a novel viewpoint, against a cluttered background, and with partial occlusions. The latter, which is also known as *category-level* or *generic* object recognition [72], is the much more challenging problem of recognizing any instance of a particular general class such as 'cat', 'car', or 'bicycle'.

While instance recognition techniques are relatively mature and are used in commercial applications, such as Photosynth [90], generic category (class) recognition is still a largely unsolved problem. Visual category recognition is an extremely challenging problem; no one has yet constructed a system that approaches the performance level of a two-year-old child [95]. However, the progress in the field has been quite dramatic, if judged by how much better today's algorithms are compared to those of a decade ago, especially because of the Convolutional Neural Network features which will be introduced next.

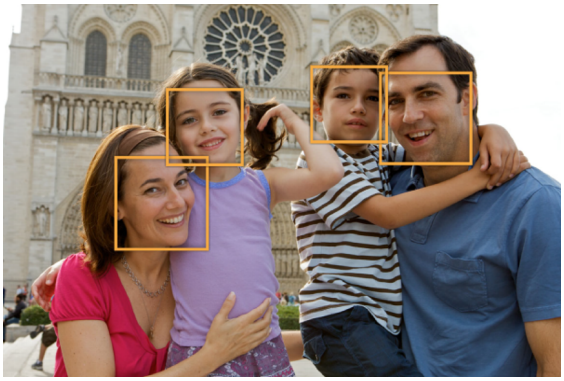


Figure 2.10: Face detection on a group portrait.
Figure courtesy: https://developer.apple.com/library/content/documentation/GraphicsImaging/Conceptual/CoreImaging/ci_detect_faces/ci_detect_faces.html (accessed May 15, 2017).

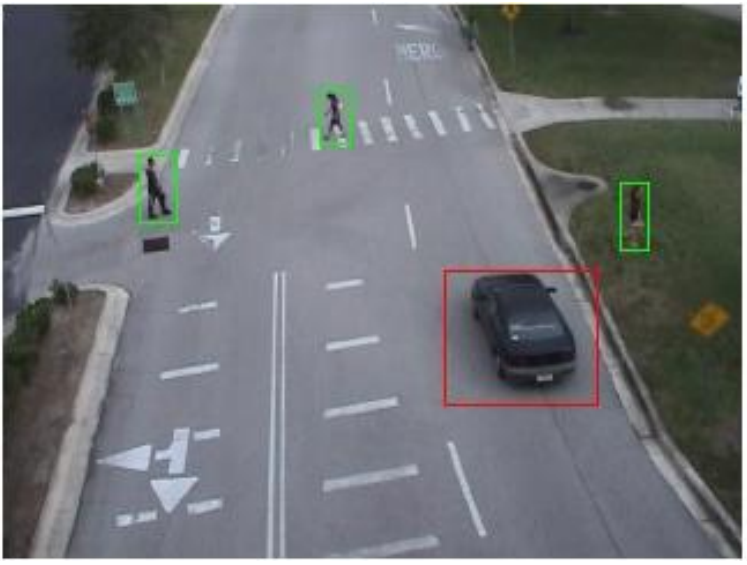


Figure 2.11: Pedestrian and vehicle detection. Figure courtesy: [42]

2.2.1 Convolutional Neural Network

Convolutional Neural Networks (CNN) are a special class of artificial neural networks which are inspired from the visual cortex of animals. Regular artificial neural networks receive an input (a single vector), and transform it through a series of hidden layers. Each hidden layer is made up of a set of neurons. Neurons in a single layer function completely independently without sharing any connections. The last fully-connected layer is called the 'output layer' and in classification settings it represents the class scores.

CNNs, on the other hand, typically have three kinds of layers - *convolutional layer*, *pooling layer*, and *fully-connected layer*:

1. The *Convolutional (CONV) layer* is the core building block of a convolutional network and does most of the computational heavy lifting. The CONV layer's parameters consist of a set of learnable filters. Every filter is small spatially (along width and height), but extends through the full depth of the input volume. For example, a typical filter on a first layer of a ConvNet might have size 5x5x3 (i.e. 5 pixels width and height, and 3 because images have depth 3, corresponding to the three color channels - *red*, *green* and *blue*). First, we slide (more precisely, convolve) each filter across the width and height of the input volume and compute dot products between the entries of the filter and the input at any position. As we slide the filter over the width and height of the input volume we will produce a 2-dimensional activation map that gives the responses of that filter at every spatial position. Intuitively, the network will learn filters that activate when they see some type of visual feature such as an edge of some orientation or a blob of some color on the first layer, or eventually entire honeycomb or wheel-like patterns on higher layers of the network. Now, we will have an entire set of filters in each CONV layer, and each of them will produce a separate 2-dimensional activation map. We stack these activation maps along the depth dimension and produce the output volume.

2. The *Pooling layer* operates independently on every depth slice of the input and resizes it spatially, using the MAX operation. The function of the pooling layer is to progressively reduce the spatial size of the representation to reduce the amount of parameters and computation in the network, and hence to also control *overfitting*.

3. *Fully-connected layer*: Neurons in a fully connected layer have full connections to all activations in the previous layer, as seen in regular neural networks.

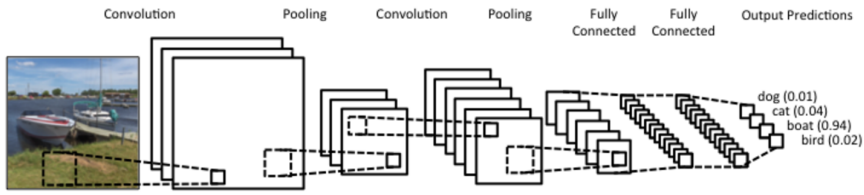


Figure 2.12: Convolutional Neural Network architecture. Figure courtesy: https://www.clarifai.com/technology (accessed May 15, 2017).

Fig 2.12 shows a sample CNN architecture. The key distinguishing feature of the CNN is that they make the explicit assumption that the inputs are images, which allows us to encode certain properties into the architecture. While regular Neural Nets don't scale well to full images, CNNs through the use of the CONV and pooling layers, offer three main benefits:

1. *Local Connectivity*: When dealing with high-dimensional inputs such as images, it is impractical to connect neurons to all neurons in the previous volume. Instead, we will connect each neuron to only a local region of the input volume. The spatial extent of this connectivity is a hyperparameter called the receptive field of the neuron. The extent of the connectivity along the depth axis is always equal to the depth of the input volume. It is important to emphasize again this asymmetry in how we treat the spatial dimensions (width and height) and the depth dimension: The connections are local in space (along width and height), but almost always full along the entire depth of the input volume.

2. *Parameter Sharing*: A parameter sharing scheme is used in Convolutional Layers to control the number of parameters. It turns out that we can dramatically reduce the number of parameters by making one reasonable assumption: That if one feature is useful to compute at some spatial position (x,y), then it should also be useful to compute at a different position (x2,y2). In other words, denoting a single 2-dimensional slice of depth as a depth slice, we are going to constrain the neurons in each depth slice to use the same weights and bias.

3. *Pooling*: Pooling simplifies the output of convolutional layers by non-linear down-sampling of the information involved in each local region. The intuition is that once a feature has been found, its exact location is no longer important.

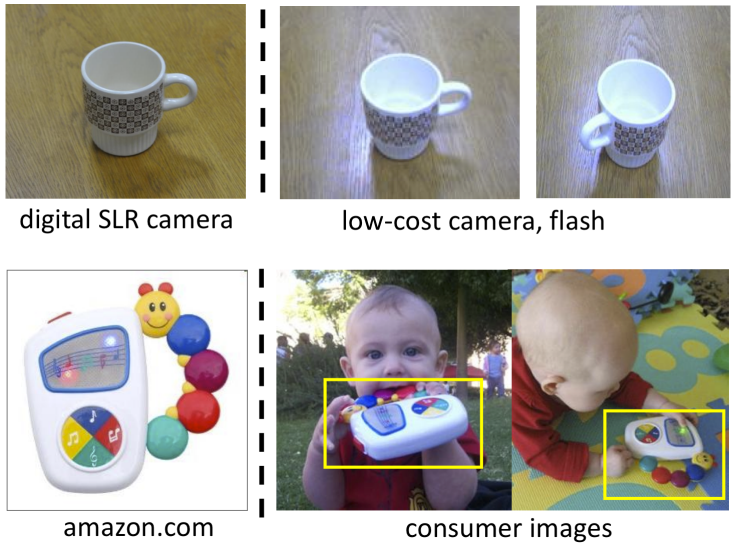


Figure 2.13: Example of domain shift in vision
Figure courtesy: https://cs.stanford.edu/ jhoffman/domainadapt/ (accessed May 15, 2017).

The activations of the penultimate fully connected layer can be used as generic features for a wide variety of tasks. In this thesis, we use these activations as features for recognizing animals in wildlife documentaries. In particular, the CNN architecture we use is the VGG CNN-M-128 architecture of [14], which is trained on 1,000 object categories from ImageNet [19] with roughly 1.2M training images.

2.2.2 Domain shift

Images depicting the same object are often very dissimilar in different domains. This phenomenon is called *domain shift*, and it could arise due to various factors such as intra-category variation, object location and pose, view angle, resolution, motion blur, scene illumination, background clutter, camera characteristics etc. Figure 2.13 shows some examples of domain shift.

Machine learning systems are often trained in controlled settings (for example, with objects localized by bounding boxes, or with objects clearly in focus) and then deployed in the wild. When the *source domain* in which the models are

trained is significantly different from the *target domain* in which the models are deployed and tested, the performance is hurt. To address this problem, *domain adaptation algorithms* [83, 7, 35, 29, 96] are developed to transfer knowledge from visual recognition systems trained on some available labeled data to the real world of natural images. The next chapter discusses one such algorithm that learns a model from images in a labeled external dataset called ImageNet, and adapts it to a target dataset of wildlife documentaries.

2.3 Natural Language Processing

This section covers key linguistic concepts used in this thesis. In particular, we cover three tasks namely *named entity recognition*, *coreference resolution* and *entity linking*. These three tasks together compose the *entity analysis stack* for language. For a more comprehensive review on these topics, we refer the reader to [43, 65].

2.3.1 Named Entity Recognition

Information extraction (IE) is the process of turning unstructured information embedded in texts into structured data, for example, distilling information like names, dates and amounts from naturally occurring text. This is an effective way to automatically populate the contents of a relational database.

Named Entity Recognition (NER) refers to the combined task of finding spans of text that constitute proper names and then classifying the entities being referred to according to their type. While early work [15] formulates Named Entity Recognition (NER) as recognizing 'proper names' in general, the scope has been widened since. Temporal expressions and some numerical expressions (i.e., money, percentages, etc.) may be considered as named entities in the context of the NER task. Named entities may also include 'natural kind terms like biological species and substances' [65]. Certain hierarchies of named entity types have been proposed in the literature, for example the BBN categories, [10]. Interestingly, this list also contains '*animals*' as one of the categories.

Alfonseca and Manandhar [2] define NER as the task of finding and classifying objects that are of interest to us. The need for precise NER tools has lead to the development of several domain-specific approaches. For example, in the biomedical domain, several methods have been proposed to recognize gene or protein names, diseases, drugs etc. [56].

Having located all of the mentions of named entities in a text, it is useful to link, or cluster, these mentions into sets that correspond to the entities. This is the task of coreference resolution, which is introduced next, and is also an important component in IE.

2.3.2 Coreference Resolution

Coreference resolution is the task of finding expressions in a text that refer to the same entity, i.e. finding expressions that *corefer*. We call the set of coreferring expressions a *coreference chain*. For example, consider the passage below:

> Victoria Chen, Chief Financial Officer of Megabucks Banking Corp since 2004, saw her pay jump 20%, to $1.3 million, as the 37-year-old also became the Denver-based financial-services company's president. It has been ten years since she came to Megabucks from rival Lotsabucks.

In processing the above passage, a coreference resolution algorithm would need to find four coreference chains:

1. {*Victoria Chen, Chief Financial Officer of Megabucks Banking Corp since 1994, her, the 37-year-old, the Denver-based financial-services company's president, She* }
2. { *Megabucks Banking Corp, the Denver-based financial-services company, Megabucks* }
3. {*her pay*}
4. {*Lotsabucks*}

Coreference resolution thus requires finding all referring expressions in a discourse, and grouping them into coreference chains. A closely related task is *pronominal anaphora resolution*. This is the task of finding the antecedent for a single pronoun; for example, given the pronoun *her*, our task is to decide that the antecedent of *her* is *Victoria Chen*. Thus pronominal anaphora resolution can be viewed as a subtask of coreference resolution.

2.3.3 Entity Linking

Entity linking, named entity linking (NEL), named entity disambiguation (NED), named entity recognition and disambiguation (NERD) is the task of determining

Figure 2.14: Example of entity linking: phrases referring to celestial objects are in blue and underlined, while the arrows point to the corresponding pages in Wikipedia.
Figure courtesy: Microsoft Cognitive services - Entity Linking Intelligence Service: https://www.microsoft.com/cognitive-services/en-us/entitylinking-api/documentation/overview (accessed May 15, 2017).

the identity of entities mentioned in text. It grounds entity mentions to their corresponding node in a Knowledge Base. Figure 2.14 shows an example of entity linking, where phrases referring to celestial objects are disambiguated, using Wikipedia[2] as the knowledge base.

Entity linking is useful wherever it is necessary to compute the direct reference to people, places, organizations or other objects of interest, rather than potentially ambiguous or redundant character strings [38]. In the finance domain, entity linking can be used to link textual information about companies to financial data, for example, news and share prices [64]. It can also be used in search, where results for named entity queries could include facts about an entity in addition to pages that talk about it [12].

Chapter 5 covers all the three tasks of the entity analysis stack, namley (i) named entity recognition, (ii) coreference resolution and (iii) entity linking.

2.4 Conclusions

This chapter provided brief overviews of fundamental concepts that are essential for a deeper understanding of the rest of this thesis. We started with the basics of statistics and machine learning concepts that will be used throughout. This included probability theory, Gaussian distributions, and a collection of

[2]https://www.wikipedia.org

probabilistic graphical models, together with their inference algorithms. We also presented a brief outline of the Expectation-Maximization algorithm.

Having covered some of the core machine learning concepts, we moved on to computer vision concepts such as object detection and recognition. In this context, we also sketch an overview of the powerful convolutional neural network, that has been revolutionalizing computer vision tasks.

Next, we touched upon some basics of natural language processing, with emphasis on the three tasks constituting the entity analysis stack - named entity recognition, coreference resolution and entity linking.

Building on some of the concepts described here, we proceed to address the first set of research questions put forth in Chapter 1. In the next chapter, we study image representations and object recognition models and apply them to the task of wildlife recognition in nature documentaries.

Chapter 3

Exploiting Labeled External Data for Weakly Supervised Wildlife Recognition

We propose a weakly supervised framework for domain adaptation in a multi-modal context for multi-label classification. This framework is applied to annotate objects such as animals in a target video with subtitles, in the absence of visual demarcators. We start from classifiers trained on external data (the source, in our setting - ImageNet), and iteratively adapt them to the target dataset using textual cues from the subtitles. Experiments on a challenging dataset of wildlife documentaries validate the framework, with a final F_1 measure of approximately 70%, which significantly improves over the results of a state-of-the-art approach, that is, applying classifiers trained on ImageNet without adaptation. The methods proposed here take us a step closer to object recognition in the wild and automatic video indexing. This chapter was published as:

Venkitasubramanian, A. N., Tuytelaars, T., and Moens, M.-F. Wildlife recognition in nature documentaries with weak supervision from subtitles and external data. Pattern Recognition Letters. Elsevier 81 (2016), 63–70.

These are **south american sea lions** off the coast of patagonia.
They can't give birth while swimming, as **whales** and **dolphins** do, but have to come ashore.
And here, in dense groups, moving awkwardly between land and sea,
they're a great temptation to any hunter that can reach them

Figure 3.1: An example of a frame with the corresponding subtitle.

3.1 Introduction

The dawn of the information age has seen tremendous growth in data especially in videos, making it increasingly challenging to facilitate quick and easy access to the relevant content. Currently, retrieval of 'relevant' videos is mostly based on user-tags. Not only are these tags often assigned in an ad-hoc manner, the process of acquiring them is also very cumbersome. Searching within the video to identify a particular segment is even more difficult, since user tags are usually not available at such fine level of detail. So, one has to manually scan the video to find a certain interesting segment.

One possible solution is to automate the indexing process, by recognizing objects or actors shown in the video and then assigning labels. The subtitles or transcripts often present in a video provide cues to derive these labels [5, 71, 70]. However, solutions proposed in the literature use a visual demarcator such as a bounding box obtained from a face detector. Moving on to the problem of recognizing animals in wildlife documentaries [23], with the current state-of-the-

art, it is not feasible to train a sufficiently accurate animal detector, since the variety within the bounding boxes is too large. Acquiring these bounding boxes by hand is tedious. Therefore, unlike [23], we are interested in a more realistic scenario where the bounding boxes are not available. In the absence of bounding boxes, the problem becomes much more challenging due to the following key issues - First, the presence of an animal is not known. Second, if the frame has animals, there could be multiple animals of possibly different species. Third, there are no ready examples that indicate with a reasonable confidence that a name-animal pair is linked. Fourth, isolating multiple animals cannot be easily done. Further, in this context, subtitles only provide weak cues, as they are not meant to describe the image content but rather give additional information to the viewer. This is in contrast to the body of work on using image captions or video descriptions [44, 28, 36], where the two modalities, namely vision and text, are much closer to each other.

In this chapter, we propose a weakly supervised approach to accurately associate animals in the video with their names in subtitles in order to assign tags or labels to video frames. We approach this as a multi-label classification problem using cross-modal data. We start from classifiers trained on external data (the source, in our setting is ImageNet [19]) and iteratively adapt them to the target dataset, using textual cues from the subtitles. In particular, we exploit the co-occurrence of animal mentions (and their co-referring expressions) in the subtitles with the animals (in their natural habitat) shown in the video to derive the correct labels. We experiment with a series of wildlife documentary videos with subtitles, from the British Broadcasting Corporation (BBC).

Figure 3.1 shows a video key frame together with the subtitle. Our approach of annotating the animals in this key frame is as follows: First, from the subtitle, we observe that the frame could contain a sea lion, or whale, or dolphin, or their combinations, or possibly no animal. We assume that if an animal is present in the video, it is also mentioned in the subtitle (or at least the subtitle contains a co-referent to it). We checked this assumption, and found out it was violated only in two key frames in our corpus. Therefore, in this example, we are interested in three binary classifiers (that indicate presence or absence) one for each possible animal - sea lion, whale and dolphin. Since we do not have reliable examples in our dataset that indicate a link between a name and an animal, we rely on an external dataset such as ImageNet to learn what these three animals look like. Then, we apply these classifiers to our data. However, as we see in Section 3.6, a direct application of the classifiers yields poor results, as the data distribution in the test (target) domain is very different from that of the training (source) domain [48]. Therefore, we propose to adapt the classifier learned on ImageNet to our dataset in an iterative manner. The basic idea of the adaptation is to exploit the co-occurrence of visually similar

patterns (in the target dataset) with the names in the subtitles. To be able to count co-occurrence of the visually similar patterns with the text, we need a mechanism for grouping visual patterns. An obvious choice would be clustering, but clustering of these frames will be extremely noisy (as we show experimentally in Section 3.6). Therefore, we propose an alternative.

Li et al. [57] have shown that the Convolutional Neural Net (CNN) features (i.e., activations of a fully connected layer of a pre-trained Convolutional Neural Network) used here have two interesting properties: Firstly, the features preserve their essence even after binarization. Second, in the context of pattern mining, Li et al. [57] have shown that for an image (patch), "the discriminative information within its CNN activation is mostly embedded in the dimension indices of the k largest magnitudes", and that these dimension indices of CNN activations can be treated as distinct items of an itemset. This means that CNN activations have the property that they are independent along the dimensions. We argue that these properties facilitate not only pattern mining of images as done in [57], but also allow *individual features (i.e, CNN activations) to be viewed as distinct elements depicting the existence (or non-existence) of some aspect of the image*. We can, therefore, represent an image with binarized CNN activations, and think of them as indicating the presence or absence of some aspect of the image. This is an intuitively appealing representation - using this representation, we can measure how the presence (or absence) of an animal label contributes to the presence (or absence) of a visual feature. This is measured by the probability of the feature given the animal name, initially using an external labeled dataset. Further, the independence property of the CNN features allows us to combine the probabilities of different features for the animal name in a naive Bayes construction to obtain the likelihood of the name for the frame. In turn, the likelihoods of the names for the frame can be used to re-estimate the probabilities of different features for the animal name, effectively adapting to the target data. The process continues until convergence.

The rest of this chapter is organized as follows: Section 3.2 discusses related work. Section 3.3 provides the background. Section 3.4 describes the general framework. Section 3.5 provides the implementation details. Section 3.6 discusses the experiments and Section 3.7 concludes the chapter.

3.2 Related work

To the authors' knowledge, the problem of aligning animals from videos with their mentions in subtitles has not been studied apart from [23].

Animals are among the most difficult objects to recognize in images and videos, mainly due to their deformable bodies that often self occlude and the large variation they pose in appearance and depiction [6, 1]. Additionally, in the natural habitat, there are challenges due to camouflage and occlusion by flora. One of the earliest works on recognition of animals was that of Schmid [84], wherein models were constructed using Gabor-like filters and tested on different classes of animals with complex texture. Later, Ramanan et al. [75] proposed models to recognize animals using the shape and texture information in videos, built from a collection of segmented images. Berg and Forsyth [6] used textual and other cues such as color, texture and shape to generate visual exemplars of various classes of animals. Apart from these works that focus specifically on animals, there is a large literature on generic object detection. These methods are often evaluated on the Pascal VOC challenge dataset [26] which among its 20 classes also includes 6 classes of animals such as cats, dogs, cows and horses. There are also datasets that focus on animals such as Caltech UCSD Birds [99] and Stanford Dogs [46]. In this work, we propose a rather generic framework using the features of [14], which are activations of a convolutional neural network, as pioneered in [53].

Recently, there has also been some work on alignment across modalities for recognizing people [71, 70, 37]. These approaches rely on the use of a face-detector. While there are face detectors available with reasonable accuracy, there are no such detectors that allow localizing animals. In fact, not being able to localize the animals complicates the problem in multiple ways. Not only does background information and image clutter affect the visual descriptors when bounding boxes are unavailable, but also the many images that do not contain an animal at all can no longer be rejected upfront.

There has also been considerable interest in sentence/caption generation from images [44, 28, 36]. These approaches are not directly applicable to our setting: first, we have too few data to train similar models. Second, in our context, the subtitles and the visuals are not parallel, but complementary. For example, often a few animals are mentioned in the text, while the connected frame only shows one of them. The connection between the vision and the text is therefore much weaker.

This work draws on the principles of domain adaptation. Most works on domain adaptation are studied in the textual domain [48, 17, 66]. Lately, domain adaptation has also been gaining interest in computer vision [7, 35, 29]. Domain adaptation in an iterative context using a naive Bayes classifier combined with an EM algorithm is also seen in [66]. In [66], text classification is performed in a multi-class setting. However, since documents and images can belong to multiple classes simultaneously, we address this problem from the perspective of multi-label classification.

A common strategy to address the classification task is to pre-train on a large dataset such as ImageNet, and then fine-tune the weights (of possibly just the high-level layers). This method is not feasible in our setting due to the large variety and lack of sufficient examples in our dataset.

The key contributions of this chapter are as follows:

1. We propose an iterative framework for domain adaption in a multi-modal context for multi-label classification.
2. Exploiting two interesting properties of CNN features, namely 1) the features preserve their essence even after binarization and 2) they can be treated independently along the dimensions, we propose a feature transformation that allows us to split an image into components, and represent the image in terms of presence or absence of components. This transformation is beneficial since it allows association of the presence (or absence) of a component with the class labels, avoiding the need for the object detection step.

3.3 Background

We have a wildlife documentary with subtitles. On the visual side, we derive key frames $\mathbf{F} = \{f_1, f_2 \dots f_q\}$ from which we extract visual features with a suitable representation $\mathbf{A} = \{\mathbf{a_1}, \mathbf{a_2} \dots \mathbf{a_q}\}$. In general, these key frames may or may not contain animals. On the textual side, from the subtitles, we extract the sentences. From these sentences, we identify the *unique* animal mentions or animal names $\mathbf{N} = \{n_1, n_2 \dots n_p\}$.

Associated with every frame $f_i, 1 \leq i \leq q$, we have a set $\mathbb{N}_i \subset \mathbf{N}$ of possible animal names derived from 5 subtitles to the left and right of the frame. The set $\mathbb{N}_i$ refers to the set of unique animal names derived from their mentions and coreferences in the subtitles. It is possible that the frame has some or all or none of the animals in $\mathbb{N}_i$. Corresponding to every name $n_l \in \mathbb{N}_i$, we have a binary label y_l indicating the presence or absence of n_l. Our objective is to find the most likely value of y_l corresponding to name n_l for every frame.

The problem of associating names to frames with manually annotated bounding boxes has been studied in [23]. As a baseline, we start with a straightforward extension of the same approach to entire frames. The basic idea is as follows - Group visual features representing frames across the whole video with a standard clustering approach such as k-means clustering. Start with an initial assumption that all the unique names are equally likely for every cluster. Iteratively refine using an EM algorithm, the likelihood of the names for the clusters, based on

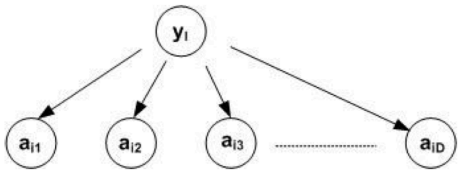


Figure 3.2: Generative model: the binary label y_l corresponding to name n_l generates the feature vector.

the co-occurrence frequency of the animal mentions with the elements of these clusters. Using the likelihoods, assign the best mapping between the animal names and the frames.

While good results were obtained with this approach when bounding boxes were available [23], applying it at the frame level is challenging due to the following key issues - first, clustering of the raw frames will be extremely noisy due to the parts of frame that do not contain animals. Note that a fuzzy-c-means clustering instead of the hard clustering will not suffice to overcome this problem. In fact, with a soft clustering, the noise from one cluster may get propagated to the other clusters. Second, it is not known if the frame contains any animal at all. Using the subtitle connected to the frame, one might conclude that the frame contains a certain animal, while the frame may in reality contain none. Under these circumstances, there are no good seed examples which indicate the possible visual representation of an animal. In the next section, we present a novel framework addressing these challenges.

3.4 General Framework

Our objective is to find the most likely value of y_l for every $n_l \in \mathbb{N}_i$ assigned to frame f_i, where $y_l = 0$ indicates the absence of the name n_l in that frame, while $y_l = 1$ indicates the presence of name n_l. Our approach is to train and iteratively adapt $|\mathbf{N}|$ classifiers, one for each name $n_l \in \mathbf{N}$. The rest of this section describes the procedure for each classifier.

3.4.1 Generative Model

The probabilistic generative model for the data is shown in figure 3.2. We assume that every frame f_i is generated according to a probability distribution defined by a set of parameters θ_l, governing the label y_l. The likelihood of a

frame is

$$p(f_i|\theta_l) = p(\mathbf{a_i}|\theta_l) = \sum_{y_l \in \{0,1\}} p(\mathbf{a_i}|y_l;\theta_l) * p(y_l) \tag{3.1}$$

The above equation involves the term $p(\mathbf{a_i}|y_l;\theta_l)$ which denotes the probability of generating the frame given the label. In sub-section 3.4.2, we describe how this term is computed, the parameter θ_l is defined in sub-section 3.4.3.

The prior $p(y_l)$ allows to bring in other information, for example, dependencies based on picturedness [70] from text analysis, or background knowledge about the documentary (for example, likelihood of tigers might be low in a documentary about Africa). For simplicity, we use an uninformed prior. So, $p(y_l = 0) = p(y_l = 1)$. Eq. 3.1 then reduces to

$$p(f_i|\theta_l) = p(\mathbf{a_i}|\theta_l) \propto \sum_{y_l \in \{0,1\}} p(\mathbf{a_i}|y_l;\theta_l) \tag{3.2}$$

Using Eq. 3.2, likelihood of all the data is

$$p(\mathbf{A}|\theta_l) \propto \prod_{f_i \in \mathbf{F}} \sum_{y_l \in \{0,1\}} p(\mathbf{a_i}|y_l;\theta_l) \tag{3.3}$$

3.4.2 Naive Bayes Model

The CNN features that we use here have properties that allow them to be treated along the dimensions independently [57]. This allows us to make the standard naive Bayes assumption. The key idea here is that rather than viewing the frame in its entirety, the frame can be viewed as a collection of D features. Then, the term $p(\mathbf{a_i}|y_l;\theta)$, of Eq. 3.1 can be estimated as follows:

$$p(\mathbf{a_i}|y_l;\theta_l) = p(< a_{i1}, a_{i2}, \ldots a_{iD} > |y_l;\theta_l) = \prod_{v=1}^{D} p(a_{iv}|y_l;\theta_l) \tag{3.4}$$

Next, we describe how the probabilities $p(a_{iv}|y_l;\theta_l)$ of individual features for the label are computed.

3.4.3 Binarization

Yet another interesting property of the CNN features is that they can be binarized without losing the essence [57]. This property can be exploited to compute the probabilities of the features $p(a_{iv}|y_l;\theta_l)$. We make use of the

fact that visually similar patterns co-occur with the same name. Instead of clustering, as done in [23] for bounding boxes and in our baseline for frames, we simply binarize the CNN features along each dimension, by splitting mid-way[1] between the minimum and maximum values of each dimension, over the entire data. The intuition behind the binning is as follows: we can represent an image with binarized CNN activations[2], and think of them as indicating the presence or absence of some aspect of the image. The binarization is intuitively appealing because with this transformation, it is easy to infer the association between the presence (or absence) of a feature and the presence (or absence) of a name.

$$p(a_{iv}|y_l;\theta_l) = p(b_v|y_l) \tag{3.5}$$

where $b_v \in \{0,1\}$ is the bin to which a_{iv} belongs.

The parameter θ_l is a collection of bin probabilities $p(b_v|y_l)$ for name n_l and bin b_v along dimension v.

3.4.4 Expectation-Maximization

For every bin b_v along dimension v and label y_l for name n_l, the parameters are initially estimated from an external reference dataset with labeled images

$$p(b_v|y_l) = \frac{freq(b_v, y_l)}{freq(b_v, y_l) + freq(\overline{b_v}, y_l)} \tag{3.6}$$

where $\overline{b_v}$ is the one's complement of b_v.

In the E-step, we estimate the posterior probability of the class labels, $p(y_l|\mathbf{a_i};\theta)$ by using Bayes' rule and applying a normalization.

$$p(y_l|\mathbf{a_i};\theta_l) = \frac{p(y_l)\prod_{v=1}^{D} p(a_{iv}|y_l;\theta_l)}{p(y_l)\prod_{v=1}^{D} p(a_{iv}|y_l;\theta_l) + p(\overline{y_l})\prod_{v=1}^{D} p(a_{iv}|\overline{y_l};\theta_l)} \tag{3.7}$$

Where $\overline{y_l} = 0$ if $y_l = 1$ and vice versa. Using Eq. 3.5, Eq. 3.7 can be written as follows:

$$p(y_l|\mathbf{a_i};\theta_l) = \frac{p(y_l)\prod_{v=1}^{D} p(b_v|y_l)}{p(y_l)\prod_{v=1}^{D} p(b_v|y_l) + p(\overline{y_l})\prod_{v=1}^{D} p(b_v|\overline{y_l})} \tag{3.8}$$

[1]We experimented with two alternatives to this equal width approach: 1) An equal frequency approach with a correction to ensure that if more than 50% of the values along a dimension are 0 (since we are dealing with sparse matrices), they should all belong to the same bin and 2) A rank-based approach where we set the r highest values along each dimension to 1 and the rest to 0. We experimented with different values of r and found that the equal width approach performed better than the equal frequency and rank-based approaches.

[2]We show in Section 3.6, that the binarization of features does not have a significant impact on the classification accuracy.

Algorithm 1: The iterative framework to identify the animals in a frame.

Input : Labeled set of images from ImageNet
Frames of the target dataset $\mathbf{F}$
Possible names $\mathbb{N}_i$ for each frame f_i

$\mathbf{N} = \cup_i \mathbb{N}_i$
for every name $n_l \in \mathbf{N}$ **do**
  Initialize $p(b_v|y_l)$ from ImageNet, using Eq. 3.6
  while likelihood measured by Eq. 3.3 increases **do**
    /* E-Step: */
    for every frame $f_i \in \mathbf{F}$ **do**
      Estimate $p(y_l|f_i;\theta_l)$, using Eq. 3.8
    /* M-step: */
    for every bin b_v along dimension v **do**
      Re-estimate $p(b_v|y_l)$ using Eq. 3.9
for every frame f_i **do**
  for every name $n_l \in \mathbb{N}_i$ **do**
    Choose the label $y_l = \text{argmax}_{y_l}\, p(y_l|f_i)$

Output : Most likely values of y_l for every frame f_i

where b_v is the bin to which a_{iv} belongs.

In the M-step, new classifier parameters, θ_l, are re-estimated based on the current values of $P(y_l|\mathbf{a_i};\theta_l)$ as follows.

$$p(b_v|y_l) = \frac{\sum_i p(y_l|\mathbf{a_i};\theta_l) * m(b_v, \mathbf{a_i}, n_l)}{Z} \tag{3.9}$$

where $m(b_v, \mathbf{a_i}, n_l)$ is 1 if a_{iv} belongs to bin b_v and name n_l occurs with frame f_l, and 0 otherwise. Z is a normalization constant to ensure $p(b_v|y_l) + p(\overline{b_v}|y_l) = 1$. These last two steps are iterated until convergence. Upon convergence, for every name n_l, the most likely label $y_l = 0$ or 1 is assigned to every frame. With this framework, it is possible that $y_l = 0;\ \forall n_l \in \mathbb{N}_i$ for a certain frame f_i. In that case, it will be predicted that the frame has no animal. This is interesting because there will be several key frames that do not contain any animal. The steps above are summarized in Algorithm 1.

3.5 Implementation Details

This section describes the pre-processing of the textual and visual data, and the learning of animal classifiers from an external dataset.

3.5.1 Pre-processing of the Textual and Visual Data

Pre-processing of the vision

- Extracting keyframes: To analyze the video, shot cut detection and keyframe extraction are done using [40].
- Extracting features: Visual features are extracted using the powerful Convolutional Neural Networks (CNN) [53], which are deep structures comprising several layers of feature extractors. We used the MATLAB interface of VGG-Net [14] with precompiled MEX files and models for computing the ConvNet features. In particular, we use the CNN-M-128 architecture, which is trained on 1,000 object categories from ImageNet [19] with roughly 1.2M training images. With this representation, the activities of the penultimate layer (7th fully connected layer) are used as features. This model yielded 128 features.

  It is worthwhile to discuss the choice of our features. It is clear that pre-trained CNN features that have been responsible for state-of-the-art performance on computer vision, are suitable for our task as well. Even within the realm of pre-trained features, there are several choices available. For example, which layer of the network should the features be drawn from? We use the most popular option for the task of object recognition, which is the vector of activities of the penultimate layer of a deep CNN, learnt on a large dataset [77] such as ImageNet [19].

  Another design choice is regarding the architecture of the network. Of various architectures, such as VGG-Net's CNN-M-128, CNN-M-1024, CNN-M-2048 (with penultimate fully connected layer containing 128, 1024 and 2048 neurons respectively), which is the best for our problem? Since our method is built on a naive Bayes classifier, it is important that we have the features to be as distinct and decorrelated as possible along the dimensions, so the conditional independence assumption is not violated. In this regard, it is ideal to choose a smaller feature vector; given that all architectures CNN-M-128, CNN-M-1024, CNN-M-2048 yield roughly similar performances (mAP of 78.60, 79.91 and 80.10 respectively on the VOC dataset), CNN-M-128 is a good choice.

Pre-processing on the text side

On the text, named-entity recognition and coreference resolution are performed as described in [23].

- Named Entity Recognition: We stem the words in the subtitles and compare them against a list of animals obtained from WordNet [63] to obtain the animals mentioned in the video.
- We combine three different coreference resolvers:
  - A basic coreference resolution system: The first resolver links a mention to the immediately preceding animal mention.
  - Coreference system of Lee et al. [54]: Stanford CoreNLP was incorporated using their Java programming API[3].
  - Reconcile coreference system [93], incorporated using their Java programming API[4].

  These tools are implemented in a cumulative fashion. That is, the final output of the coreference resolution is the union of the outputs of all the methods.

Assigning possible names to frames

We mapped every frame to the five subtitles to the left and right of it. All animal names and coreferences (obtained using from the above step) in this range of subtitles were assigned to a frame. The pre-processing of text and assignment of names to frames were implemented in Java.

3.5.2 Learning from ImageNet

We use ImageNet to learn probabilities of the binarized features given the name. The process is as follows: For every unique animal name n_l, we collect a set $\mathbf{I_{n_l}}$ of images from ImageNet. The set $\mathbf{I} = \cup_{n_l \in \mathbf{N}} \mathbf{I_{n_l}}$ constitutes a dataset labeled with animals that we use for training animals classifiers.

We then train binary classifiers for each of these entities on the collection of relevant images from ImageNet, using a one-versus-rest scheme. For each unique animal mention n_l, the positive class comprises the images $\mathbf{I_{n_l}}$ containing that animal, while the negative class includes all the other images, $\mathbf{I} - \mathbf{I_{n_l}}$. On inspecting the data from ImageNet, it was found that there were very few examples with multiple species in the same image, so it is reasonable to assume that the negative class for an animal does not include that animal.

[3]https://stanfordnlp.github.io/CoreNLP/ (accessed May 17, 2017).
[4]https://www.cs.utah.edu/nlp/reconcile/ (accessed May 17, 2017).

For all the images in $\mathbf{I}$, we extract the CNN visual features trained on ImageNet [53] as before and binarize them. Once the bins have been computed, the probability of a bin b_v along the dimension v for a label y_l is estimated by counting the co-occurrence of the name with the bin, using Eq. 3.7.

The initialization described above and the iterative domain adaptation algorithm were implemented in MATLAB. The source code is made publicly available at: https://github.com/aparnavenkit/Multimodal-Domain-Adaptation.

3.6 Experiments and Results

The data used in our experiments is the DVD Great Wildlife Moments[5] from the BBC. This is an interlaced video with a duration of 108 minutes at a frame rate of 25 frames per second, and the frame resolution is 720x576 pixels. The video consists of 28 chapters and all the chapters except the ones containing just one animal are evaluated. This leaves us with chapters 14 to 28. Applying shot cut detection [40] on these chapters, we obtained 602 key frames. Of these, 302 frames had no animal. The remaining 300 contained 365 animals in total. We run our algorithm on all the 602 frames.

The subtitles are distributed throughout the video and contain a total of 7,304 words in 545 sentences. 186 animal names are mentioned in these subtitles. The distribution of animal species over the key frames is shown in figure 3.3. The number of animal mentions associated with each frame over the entire dataset is also shown in Figure 3.3. Note that based on the subtitles, there are several frames that have at least 2 names associated. On the visual side, however, there are several frames that do not contain any animals. This shows the ambiguity in text and vision.

The evaluation of the text pre-processing is as in [23]. In order to evaluate our algorithm, we first consider a set of approaches purely based on vision, using an external dataset (Section 3.6.1). Second, we consider a baseline entirely based on text (Section 3.6.2). Third, we report the results of state-of-the-art approaches based on clustering and understand their shortcomings in our scenario where bounding boxes are absent (Section 3.6.3). Next, we study the impact of binarization from two perspectives- a) as an alternative feature representation and b) as a means of grouping (Section 3.6.4). Finally, we evaluate our pipeline and show the value of the iterative learning (Section 3.6.5). Table 3.1 shows the results of our approach compared to various other approaches.

[5]https://en.wikipedia.org/wiki/Great_Wildlife_Moments (accessed May 15, 2017).

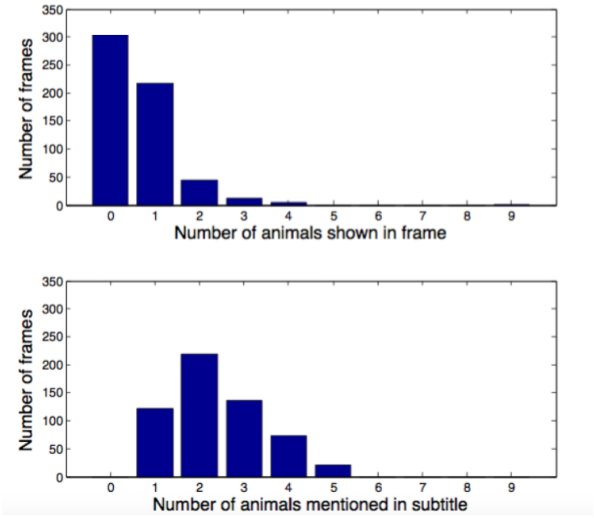


Figure 3.3: Distribution of animals over the key frames.

Precision and recall are computed over the entire dataset as follows:

$$\text{precision} = \frac{\text{number of correct guesses}}{\text{total number of guesses}} \tag{3.10}$$

$$\text{recall} = \frac{\text{number of correct guesses}}{\text{actual number of animals present}} \tag{3.11}$$

The F_1 is computed using the precision and recall as follows:

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{3.12}$$

Figure 3.4 shows an example of our approach in the realistic scenario without bounding boxes.

3.6.1 How Good is Classification Solely Based on ImageNet?

To answer this question, we consider the following approaches where a model learned on ImageNet is applied to our dataset.

- **CNN-M-128**: We deploy the CNN-M-128 architecture of [16] that was used for feature extraction. However, instead of using the activations of

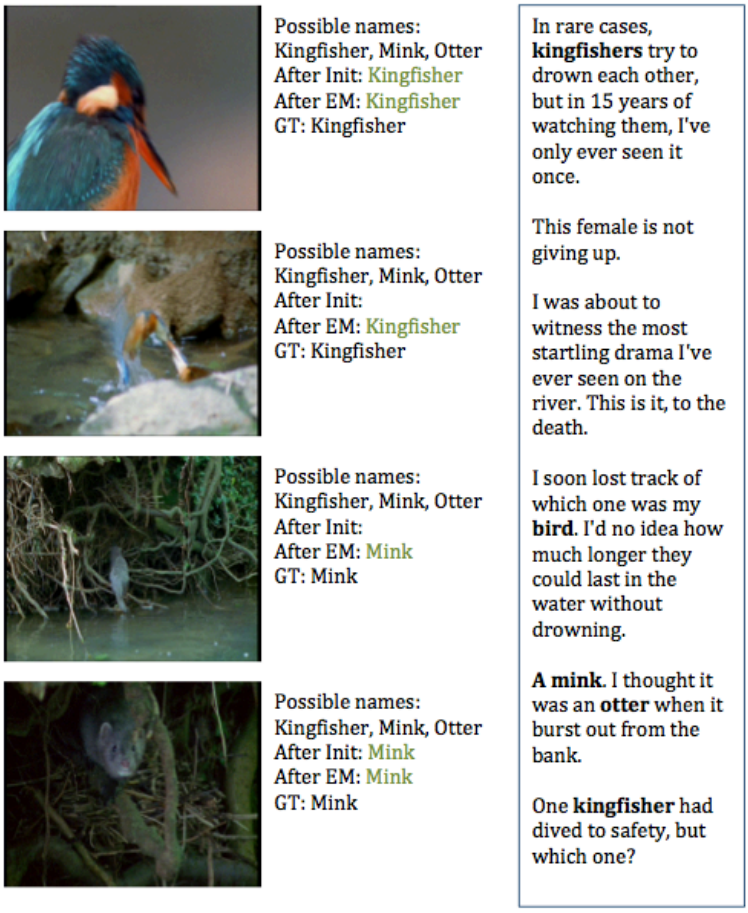


Figure 3.4: Annotating animals shown in the video key frames using the subtitle: Key frames (left), Predicted names (center), Subtitles (right); Init refers to the initialization using ImageNet and GT refers to the ground truth.

the penultimate layer, we use the probability outputs of the final layer. If the probability of a certain animal is greater than 0.5, we conclude that the animal is present in the frame. The precision and the recall of this method are rather low. The reason is largely attributed to the domain shift (Figure 3.5)- the background plays a bigger role in our video, compared to the ImageNet images where the subject is central. Moreover, the images in ImageNet are of better quality with a high resolution, while the video key frames are of lower quality. Additionally, only 15 of our 19

Method	Precision	Recall	F_1
CNN-M-128	13.1	25.8	17.3
CNN-M-128 filtered	55.0	25.8	35.1
ImageNet SVM$_{raw}$	32.9	28.9	30.8
Only text	42.5	**97.7**	59.3
Clustering + text + EM	55.7	36.4	44.0
ImageNet SVM$_{binarized}$	28.9	29.9	29.4
Binarization + text + EM	55.9	44.6	49.6
ImageNet NBC	15.7	45.5	23.4
ImageNet NBC + text	**57.6**	44.6	50.3
ImageNet NBC + text + EM	**57.3**	**88.7**	**69.6**

Table 3.1: Evaluation of the indexing of frames (in %).

names were present in the 1000 classes of ImageNet. However, the drop in recall resulting from the 4 missing classes was only 5.15%.

- **CNN-M-128 filtered**: We modified the above approach so as to exclude those animals that were not in our dataset. Although this has lead to an increase in the precision compared to the above approach, the recall remains low.

- **ImageNet SVM$_{raw}$**: Yet another simple solution to the problem of labeling is to train SVMs on labeled data, and apply it on our unlabeled dataset. Here, we train SVMs on the images extracted from ImageNet (in a one-vs-rest scheme), using the raw (non-binarized) CNN features, and test it on our dataset on the raw visual features. Note that the precision, recall and consequently the F_1 measure are quite low. Again, this is a result of the domain shift.

3.6.2 How Good is the Text?

We consider the baseline **Only text** which basically assigns all the possible names derived from 5 subtitles to the left and right of the frame. Note that the precision is quite low. However, the recall is very high; by extracting the names within this range of subtitles, almost all animal mentions were recovered. A recall of 100% is not achieved owing to two reasons - First, there were a couple of frames showing ducks, but the word duck is not mentioned in the subtitles over the entire video. Second, because we use a subtitle window of 5 subtitles, a few names are missed.

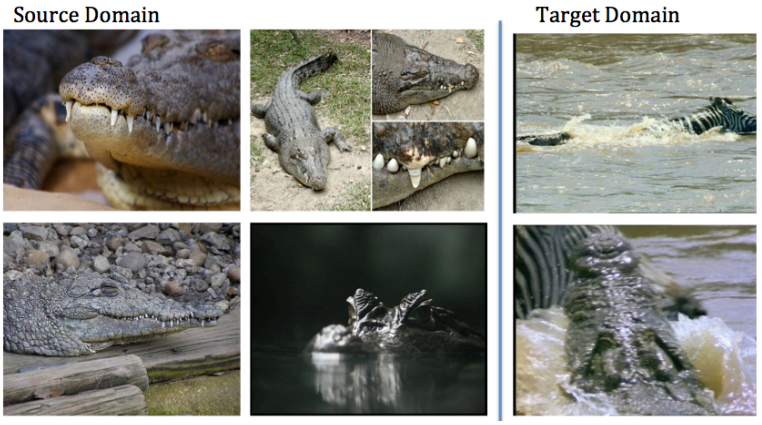


Figure 3.5: Images of crocodile from ImageNet (left) and keyframes containing crocodile (right).

3.6.3 Will Clustering-based Solutions Work?

The results of the labeling in an ideal scenario with manually annotated bounding boxes, as described in Section 3.3, using the approach of [23] with CNN features are shown in Table 3.2. *Annotation* indicates which bounding box in the video maps to which entity in the subtitle. *Frame indexing* indicates what animals shown in the frame are mapped to entities in the subtitles, considering the objects in the video frame and the entities in the subtitles as two distinct groups. The frame indexing deals with the mapping of the groups of objects in the frame to the groups of entities in the subtitles, ignoring the actual correspondence of the individual animals/entities.

We also apply the approach of [23] to entire frames rather than bounding boxes[6]. This is the baseline **Clustering + text + EM**. Here, we used k-means clustering to cluster the frames, with k set to 20, since there were 19 entities and we added 1 cluster for the background. Figure 3.6 shows some of the clusters obtained. It can be seen that the clusters are rather noisy. First, when there are multiple species in the same frame, they are forced into one cluster. For example, in the first cluster of Figure 3.6, zebra and crocodile are in the same cluster, simply because they are in the same frame. Second, even when frames with just one animal are involved, they are often grouped incorrectly.

[6]There exist methods such as [33] to propose bounding boxes. While these methods have a high recall, the precision is often not sufficient for methods such as [23] to work. We experimented with the top 1 and 2 bounding boxes per frame, and found the performance quite low.

Method	Annotation	Frame Indexing		
	F_1	Precision	Recall	F_1
Initialization	80.80	87.1	82.3	84.6
After EM	83.80	88.5	86.4	87.4
Ground truth clusters	95.1	96.6	95.2	95.9

Table 3.2: Clustering-based algorithm applied on manually annotated bounding boxes (in %).

For example, in the first cluster of Figure 3.6, hippopotamus, crocodile, and kingfisher are in the same cluster. As a result, the performance of this approach in our setting is low, especially, the recall. This is because when a frame falls into the wrong cluster, the likelihood of the associated name would be very low, based on the other elements of the cluster. Next, we show that the binarization we proposed copes with this issue.

3.6.4 What is the Impact of Binarization?

To study the impact of binarization, we consider the following baselines.

- **ImageNet SVM$_{binarized}$**: We train SVMs as in Section 3.6.1, but using the binarized features instead of the raw features. The classifiers are then applied to our dataset with binarized features. While the results of this approach are quite low, they are comparable to the case with raw features (ImageNet SVM$_{raw}$). This is consistent with the study of [57].
- **Binarization + text + EM**: In this approach, rather than clustering the entire frames, we binarize[7] features along the dimensions. We start with uniform probabilities of the binarized features for the names (instead of learning from ImageNet). These probabilities are then refined by the E and M steps denoted by Eq. 3.8 and 3.9 respectively. Note that the precision improves significantly over the clustering-based approach (Clustering + text + EM). There is also an improvement in recall. *Binarization as an approach to grouping seems better than clustering in this setup with CNN features.*

[7]While it is possible to split the data into more bins, we have empirically found that the optimal number of bins for these features is 2.

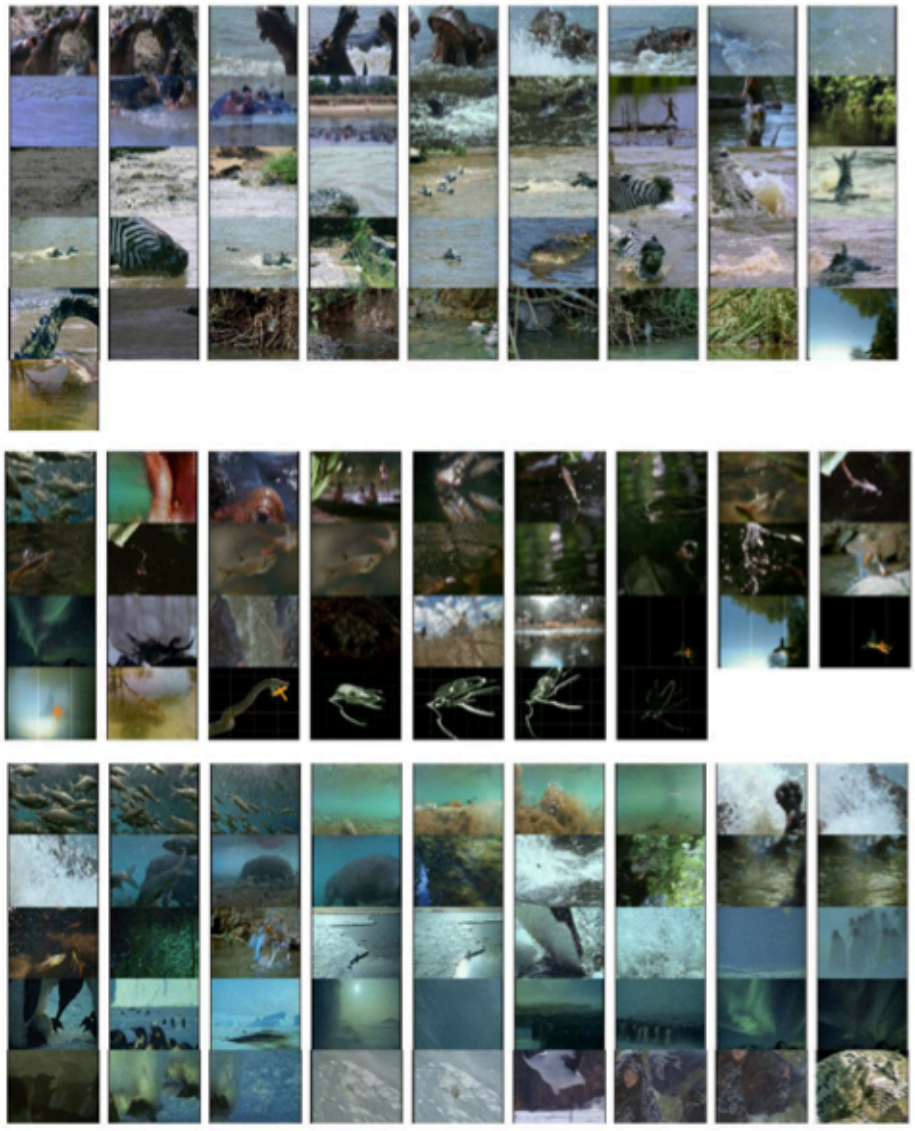


Figure 3.6: Clusters of key frames.

Figure 3.7: Examples of key frames annotated by our system compared to the ground truth annotations (GT).

3.6.5 What is the Value of the Iterative Learning?

To evaluate our pipeline, we consider the following approaches.

- **ImageNet NBC**: In this approach, we learn initial probabilities (of the binarized features) from ImageNet and combine them using a naive Bayes construction (Eq. 3.4). Textual cues are not used. It is interesting to compare the naive Bayes (binarized) with the binarized SVM. While the precision of the naive Bayes is quite low, the recall is better than that of $\text{SVM}_{binarized}$.
- **ImageNet NBC + text**: As before, we learn initial probabilities (of the binarized features) from ImageNet and combine them using a naive Bayes construction (Eq. 3.4). Further, we filter the labels by using the subtitles connected to a frame. Basically, we assign the labels to the frame only if the naive Bayes predicts the label and if the label is also present in the adjoining subtitle. Note that the precision increases significantly over the above approach when textual cues are used. This is explained as follows. *The text provides good cues about the presence of certain animals.* For instance, very often hippopotamus or crocodiles were classified as salmon, simply because of the presence of the water body in the background. Using the textual cues, it is possible to arrive at the conclusion that it is unlikely that a salmon is shown here. This increase in precision is accompanied by a small drop in recall. The reason is that in some cases, although the classifier predicted a certain label correctly, the text suggested that the label may not be relevant in the context.
- **ImageNet NBC + text + EM**: This is basically the entire pipeline. We start with classifiers trained on ImageNet and iteratively adapt them to our dataset by making use of the textual co-occurrence information. Compare these results to two other approaches

1. **Binarization + text + EM**, where we started with uniform probabilities rather than learning from ImageNet. Note the *significant increase in recall and precision when the probabilities are learned from ImageNet.* The probabilities learned from ImageNet provide a good initialization that is essential to make the method converge to a meaningful result.
2. **ImageNet NBC + text**, where we learned probabilities from ImageNet and filtered the labels using the text. Again, there is an increase in the recall because the *EM iterations adapt the classifiers trained to suit our data.*

The learning from ImageNet combined with the iterative use of textual cues that suggest the relevance of certain animals has boosted the recall significantly.

In addition to the evaluation on the entire dataset, we divided the frames into chapters and executed the pipeline on the individual chapters. The macro-average precision, recall and F_1 were 55.6%, 92.2% and 69.4%, while the micro-average precision, recall and F_1 were 58.1%, 94.3% and 71.9% respectively. These results are in line with the finding that the entire pipeline improves over each of the other methods for our documentary dataset.

Additionally, we tested the statistical significance of the results using a frame-level paired t-test and found that our method was significantly better ($p < 0.001$) than all approaches. Note that we are interested in a method that has the best performance in terms of precision and recall taken together. Figure 3.7 shows some examples of key frames annotated by our system. Particularly, even though the image with the penguins (4^{th} key frame) is hazy, this algorithm is successful in identifying the correct animal. Our algorithm is also successful in deducing that there are no animals in a frame (Figure 3.7, 3^{rd} key frame).

3.7 Summary and Conclusions

This chapter shows that *by training classifiers on an external labeled dataset, and adapting them iteratively to the target dataset, using textual cues, the accuracy of classification can be improved.* This is applied to the context of recognizing objects such as animals shown in the video with subtitles, in the absence of visual demarcators such as bounding boxes. Exploiting the synergy between the visual features, textual cues and an external dataset, the accuracy of our approach is significantly better than a) a purely vision-based approach or b) purely text-based approach or c) an approach that uses both text and vision,

but without labeled examples or d) an approach that uses both text and vision, and labeled (out-of-domain) examples, but without the adaptive learning.

In the future, we wish to apply our algorithm to other datasets for furtherance of the evaluation scope. Additionally, we would like to determine the influence of the background in the recognition of animals, to determine whether or not the background should be used. Further, we intend to filter out regions of no interest which would confuse the clustering or classification. Applying these techniques allows making videos 'searchable' by automatically indexing them. In the next chapter, we study object recognition models that can perform without external labeled examples.

Chapter 4

A Study of Image Representations and Wildlife Recognition Models

In this chapter we investigate animal recognition models learned from wildlife video documentaries by using the weak supervision of the textual subtitles. As mentioned in the earlier chapters, this is a challenging setting, since i) the animals occur in their natural habitat and are often largely occluded and ii) subtitles are to a great degree complementary to the visual content, providing a very weak supervisory signal. This is in contrast to most work on integrated vision and language in the literature, where textual descriptions are tightly linked to the image content, and often generated in a curated fashion for the task at hand. The previous chapter overcomes these challenges by leveraging external labeled data. In this chapter, we address the task without using external training data. We investigate different image representations and models, in particular a support vector machine on top of activations of a pretrained convolutional neural network, as well as a naive Bayes framework on a *'bag-of-activations'*, where each element of the bag is considered separately. This *'bag-of-activations'* paradigm allows key components in the image to be isolated, in spite of vastly varying backgrounds and image clutter, without an object detection or image segmentation step. The methods are evaluated based on how well they transfer to unseen camera-trap images captured across diverse topographical regions under different environmental conditions and illumination settings, involving a large domain shift. The work presented in this chapter is based on:

In the rivers and lakes of Africa, lives an animal which has a reputation for being the most unpredictable and dangerous of all.
Even **crocodiles** are wary.
The **hippopotamus**.

Figure 4.1: A set of frames together with the corresponding subtitles: The frames show hippos, while the subtitles mention both hippo and crocodile.

Venkitasubramanian, A. N., Tuytelaars, T., and Moens, M.- F. Learning to Recognize Animals by Watching Documentaries: Using Subtitles as Weak Supervision. In Proceedings of the EACL Workshop on Vision and Language (2017).

4.1 Introduction

It is estimated[1] that video traffic will be 82 percent of all global Internet traffic by 2020. The ubiquitousness of video on the web demands indexing tools that facilitate fast and easy access to relevant content. Traditionally, video search has been based on user-tags. However, in the recent past, research activities have been directed at automatic indexing of videos based on the content. Contributing to this goal of automatic video indexing, we focus on the problem of wildlife recognition in nature documentaries with subtitles.

As mentioned in the earlier chapters, this setup is challenging from at least two perspectives: first, from the point of view of the *content*, and second, due to the *nature of video documentaries*. As far as the *content* is concerned, we are dealing

[1] http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html (accessed May 15, 2017).

with animals shot in their natural habitat. The problem of identifying animals in videos, especially those shot in the natural habitat presents several challenges. Firstly, animals are among the most difficult objects to recognize in images and videos [1, 6]. Further, in the natural habitat, there are challenges due to camouflage and occlusion by flora. Moreover, unlike faces or cuboidal objects such as furniture, we do not have accurate detectors that can localize the animal in a frame. State-of-the-art object proposal methods such as [33, 79] yield an unacceptably low level of either recall or precision. The absence of detectors necessitates other mechanisms that allow segregation of the components of the image.

The *nature of video documentaries* presents yet another challenge. Typically, in video documentaries such as ours, the subtitles are not parallel, but complementary to the visuals (see Fig. 4.1). This is in contrast to most work on integrated vision and language in the literature, where textual descriptions are tightly linked to the image content. This means we do not have examples that can reliably tie together textual and visual entities.

While the previous chapter overcomes these challenges by leveraging external labeled data, here we study image representations and models that cope with these without the need for external training data. These include a support vector machine on top of activations of a pretrained convolutional neural network, and a naive Bayes framework on a *'bag-of-activations'*, where each element of the bag is considered separately. While the former utilizes a *global perspective* where the feature vector comprising CNN activations is viewed as one entity, the latter works on per dimension basis, allowing key components in the image to be isolated, in spite of largely varying backgrounds and image clutter, without an object detection or image segmentation step. We experiment with both continuous and discretized variants of the *'bag-of-activations'* perspective. In particular, *we investigate image representations and weakly supervised animal recognition models that can be learned without the need for bounding boxes, or curated data comprising manually annotated training examples.*

The rest of this chapter is organized as follows: Section 4.2 presents the background and related work. Section 4.3 provides the problem definition. Section 4.4 describes the image representations and animal recognition models based on CNN activations. Section 4.5 provides implementation details. Section 4.6 discusses the experiments and results. Finally, Section 4.7 provides the conclusions.

4.2 Background

Identifying animals is a well-studied topic [1, 6, 84, 75]. Recent works such as [39] and [34] advance us further and provide better insight into the problem. However, these methods are not applicable in our setting since they require extensive training data. It is important to note that in this setup, we lack sufficient reliable training data making neural network-based training impractical.

The previous chapter addresses the problem of aligning animals from videos with their mentions in subtitles. Recall that the method discussed in that chapter learns object recognition models from an external labeled dataset (ImageNet) and iteratively adapts the models to the target wildlife documentary. It has the issue that not all classes of objects can be learned from an external dataset, for instance, rare species of animals may not be found on ImageNet.

In this chapter, we investigate image representations and multi-modal animal recognition models that can cope with the lack of labeled external data, in addition to dealing with the complementarity of vision and language, and the lack of bounding boxes. Further, we study how such models trained on one dataset transfer to a different unseen domain, shot under very different conditions.

4.3 Task Definition

The task is the same as that of the previous chapter: given a wildlife documenatry with subtitles, we are interested in identifying the animals present in each keyframe. On the visual side, we derive key frames $\mathbf{F} = \{f_1, f_2 \dots f_q\}$ from which we extract visual features with a suitable representation $\mathbf{A} = \{\mathbf{a_1}, \mathbf{a_2} \dots \mathbf{a_q}\}$. Assume each feature vector has D dimensions. On the textual side, from the subtitles, we identify the *unique* animal mentions or animal names $\mathbf{N} = \{n_1, n_2 \dots n_p\}$, using a list of animal names derived from WordNet [63] as in [23].

Using the setup of [97], we associate every frame $f_i, 1 \leq i \leq q$, with a set $\mathbb{N}_i \subset \mathbf{N}$ of possible animal names derived from 5 subtitles to the left and right of the frame. The set $\mathbb{N}_i$ refers to the set of unique animal names derived from their mentions and coreferences in the subtitles[2]. It is possible that the frame has some or all or none of the animals in $\mathbb{N}_i$. Corresponding to every name $n_l \in \mathbb{N}_i$, we have a binary label y_l indicating the presence or absence of n_l. Our

[2]There remains a small percentage (2.35%) of animals not mentioned in the nearby subtitles. These will be left undetected.

objective is to find the most likely value of y_l corresponding to name $n_l \in \mathbb{N}_i$ for every frame f_i.

4.4 Image Representations Based on CNN Activations

A popular choice of visual features for object recognition is the activations of the penultimate layer of a pretrained Convolutional Neural Network. In this work, we use the VGG CNN-M-128 architecture[3] of [14], which is trained on 1,000 object categories from ImageNet [19] with roughly 1.2M training images. Within this realm, we explore two perspectives on the real-valued feature vector: (i) a *global perspective* where each feature vector is viewed as one entity, and (ii) a *bag-of-activations perspective*, where each element of the bag is considered separately. Note that in either case, the feature vectors are exactly the same; the difference only lies in whether each feature vector is treated as one entity, or as a bunch of distinct attributes or components.

The **global perspective** is by far the most commonly used [77] and fits well with a linear Support Vector Machine (SVM) classifier. For the task of object recognition, the linear SVM is typically used with the L_2 norm, and has the following objective function

$$\underset{\mathbf{w_l}}{\text{minimize}} \frac{1}{2}||\mathbf{w_l}||^2 + C\sum_i \max(1 - y_l \mathbf{w_l}^T \mathbf{a_i}, 0)$$

where $\mathbf{w_l}$ denotes the set of weights to be learned for the label y_l corresponding to name n_l, and C denotes the cost[4]. In a weakly supervised setting, these weights are learned based on the weakly associated (hence noisy) frame-name pairs $< \mathbf{a_i}, n_l >$ for all $n_l \in \mathbb{N}_i$.

An alternative to this *global perspective* is a **bag-of-activations** perspective, where each feature dimension is treated in isolation. As indicated in the previous chapter, CNN activations have two interesting properties: firstly, they can be treated independently along the dimensions and second, they preserve their essence even after binarization. We exploit the first property and use it in a naive Bayes framework. The idea of treating each component of the CNN representation individually rather than using the full feature vector in a high-dimensional space is crucial: *It brings robustness to image clutter and changing backgrounds, and helps in learning from few examples.*

[3]This model yielded 128 features.

[4]We used the Liblinear [27] toolkit, with the default setting of 1 for the cost C.

To compute the probability of a label y_l for any frame $\mathbf{a_i}$, we compute the probability of the individual features $a_{i1}, a_{i2} \dots a_{iD}$ for the label y_l.

$$p(y_l|\mathbf{a_i}) = \frac{p(y_l)\prod_{v=1}^{D} p(a_{iv}|y_l)}{Z_l} \tag{4.1}$$

Z_l is a normalization constant for the name n_l, given by

$$Z_l = p(y_l)\prod_{v=1}^{D} p(a_{iv}|y_l) + p(\overline{y_l})\prod_{v=1}^{D} p(a_{iv}|\overline{y_l}) \tag{4.2}$$

where $\overline{y_l} = 0$ if $y_l = 1$ and vice versa. $p(y_l)$ is the prior which we assume to be uninformative for simplicity. So, $p(y_l = 0) = p(y_l = 1)$.

Then, using Eq. 4.2, Eq. 4.1 can be written as follows:

$$p(y_l|\mathbf{a_i}) = \frac{\prod_{v=1}^{D} p(a_{iv}|y_l)}{\prod_{v=1}^{D} p(a_{iv}|y_l) + \prod_{v=1}^{D} p(a_{iv}|\overline{y_l})} \tag{4.3}$$

Note that for computing these posterior probabilities we rely on the probability of the individual features or attributes for a label. In our setting with complementary modalities instead of parallel data, this is particularly relevant. The absence of parallelism means that there are too few examples to learn. By assuming conditional independence of the features given the label, the naive Bayes overcomes the need for a large training set.

The second interesting property of the CNN activations is that they preserve their essence even after binarization. We investigate this further and show that not only binarization but also **discretization** of the feature vector into a larger number of bins is useful (as shown in Section 4.6). In particular, we propose to discretize the feature vector into B bins along each dimension[5]. In this chapter, we experiment with two approaches for binning the feature vector - (i) equal width and (ii) equal frequency. The equal width approach ensures that all the bins are of the same size. For example, if we are interested in 2 equal width bins, we could look at the feature vector along a dimension and set the threshold midway between the minimum and maximum values of that dimension. The values that are less than the threshold could be set to 0, while the rest are set to 1. In equal frequency binning, the threshold is set such that the number of elements in each bin is roughly the same.

This discretization is similar to the vector quantization of SIFT descriptors to obtain Bag of Visual Words (BoVW). But, while BoVW has the issue that the

[5]Discretization can also be applied to the *global representation* used by the SVM, but as shown in [97], it is particularly useful in conjunction with a naive Bayes classifier.

discretization errors can have a significant negative impact, with CNN features, *there are no strong discretization artifacts.* In fact, Li et al. [57] have shown that retaining just the values of the largest k dimensions (or even setting the values of the largest k dimensions to 1 and the rest to 0) is sufficient to capture the essence of the image.

Discretizing the feature space allows us to replace the feature a_{iv} by the corresponding bin b_v.

$$p(a_{iv}|y_l) = p(b_v|y_l) \tag{4.4}$$

where $b_v \in \{0, 1 \dots B\}$ is the bin to which a_{iv} belongs.

Eq. 4.3 can then be rewritten as

$$p(y_l|\mathbf{a_i}) = \frac{\prod_{v=1}^{D} p(b_v|y_l)}{\prod_{v=1}^{D} p(b_v|y_l) + \prod_{v=1}^{D} p(b_v|\overline{y_l})} \tag{4.5}$$

To compute the conditional probabilities $p(b_v|y_l)$ of the bin b_v given y_l, we rely on the noisy labels that can be obtained from the text. Basically we count the co-occurrence of label y_l corresponding to name $n_l \in \mathbb{N}_i$ with bin b_v relative to the total number of instances where y_l occurs in our dataset.

$$p(b_v|y_l) = \frac{freq(b_v, y_l)}{freq(y_l)} \tag{4.6}$$

4.5 Implementation Details

- The pre-processing on the vision and language were done exactly as in the previous chapter.
- For the SVM, we used the MATLAB interface of the LIB-LINEAR [27] toolkit, which outputs both predicted labels and the confidence scores for each prediction.
- For the naive Bayes on the continuous features, we used MATLAB's built-in naive Bayes classifier, which outputs both predicted labels and the confidence scores for each prediction.
- For the naive Bayes on the discretized features, we implemented a multinomial naive Bayes classifier in MATLAB.

Method	Precision	Recall	F_1
SVM	80.43	12.71	21.96
Naive Bayes	20.23	71.48	31.54

Table 4.1: Results of using the *continuous features* and applying the weak labels of our dataset.

4.6 Experiments and Results

The dataset and experimental setup are exactly the same as the previous chapter. Our dataset consists of 602 key frames and 19 species of animals.

The animal labeling is evaluated in terms of precision, recall and F_1 as indicated in the previous chapter.

The evaluation covers two aspects:

1. How well do the representation and model learned using the weak labels of our dataset perform on the same dataset? (Section 4.6.1)
2. How well do the representation and model learned using the weak labels of our dataset transfer to an external dataset shot over diverse topographical regions under different environmental conditions and illumination settings? (Section 4.6.2)

4.6.1 Animal Labeling on Wildlife Videos

Table 4.1 shows the performance of an SVM on the *global perspective* and a naive Bayes classifier on the *bag of activations* using *continuous features*. In either case, name n_l is assigned to frame $\mathbf{a_i}$ if $p(y_l|\mathbf{a_i}) > p(\overline{y_l}|\mathbf{a_i})$, that is, the probability threshold for prediction was set at 0.5. For the naive Bayes classifier, a Gaussian distribution was used to model the continuous features along each dimension. While both models do not yield adequate performance, the naive Bayes certainly does far better compared to the SVM. In this setup involving limited reliable example pairs, *it is beneficial to treat each element of the CNN representation individually rather than using the full feature vector in a high-dimensional space*. Fig. 4.2 shows the precision-recall curves of the SVM and the naive Bayes classifier. The naive Bayes is clearly better in this setup, except in the low recall / high precision region.

Closer inspection reveals that the Gaussian distribution used in the naive Bayes framework is not a good fit to the data. For example, consider Fig. 4.3. Fig. 4.3

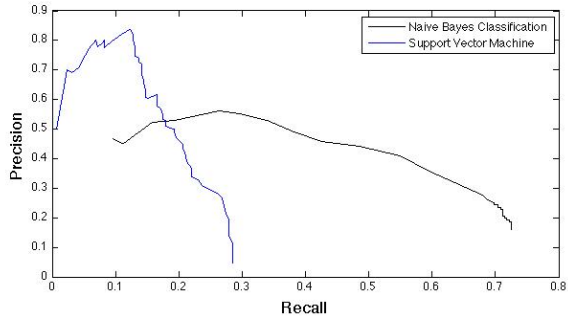


Figure 4.2: The precision-recall curves for the SVM and naive Bayes classifier shown in Table 4.1. Area under the curve is 0.1599 for the SVM and 0.3642 for naive Bayes.

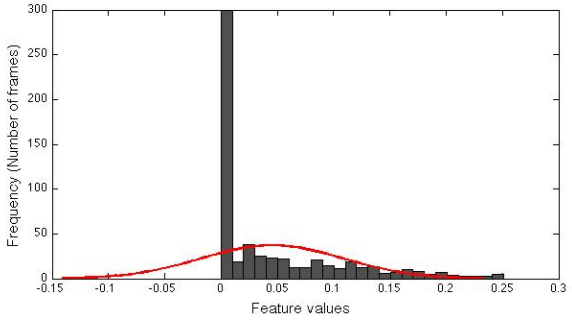


Figure 4.3: The distribution of the feature values along the first dimension: x-axis shows the range of feature values, y-axis shows the number of frames. The gray histogram shows the distribution of the feature values. The red curve is the normal distribution plotted using the mean and standard deviation along the first dimension, $\mathcal{N}(0.0454, 0.0622)$.

Method	Precision	Recall	F_1
$B = 2$	46.43	91.55	61.61
$B = 3$	46.85	94.37	62.62
$B = 4$	47.03	92.96	62.46
$B = 5$	47.18	94.37	62.91
$B = 6$	47.88	95.31	63.74
$\boldsymbol{B = 7}$	47.69	96.71	**63.88**
$B = 8$	47.45	96.24	63.57
$B = 9$	47.00	95.77	63.06
$B = 20$	46.47	95.77	62.58
B $= log_2 l$	47.47	96.71	63.68

Table 4.2: Results of using the *discretized features* using equal width discretization and applying the weak labels of our dataset.

Method	Precision	Recall	F_1
$B = 2$	48.04	92.02	63.12
$B = 3$	47.95	93.43	63.38
$B = 4$	46.99	95.31	62.95
$B = 5$	46.24	95.31	62.27
$B = 6$	45.56	96.24	61.84
$B = 7$	45.23	95.77	61.45
$B = 8$	44.93	95.77	61.17
$B = 9$	44.81	97.18	61.33
$B = 20$	43.51	97.65	60.20

Table 4.3: Results of using the *discretized features* with equal frequency discretization and applying the weak labels of our dataset.

shows the normal distribution plotted using the mean and the standard deviation along the first dimension for the entire dataset (red curve: $\mathcal{N}(0.0454, 0.0622)$). This is superimposed on the histogram of the real-valued (undiscretized) feature vector (in gray). While there are certainly other distributions (such as Poisson or Binomial) that could be used to model the data, we show that the most commonly used Gaussian clearly does not fit the data. Rather than forcing the data to fit into some distribution, we turn to a discretized setting as it allows using a simple non-parametric model. *The discretization overcomes the need to make assumptions on the data distribution, and allows approximating the density function using a histogram.*

Next, we present the results of using the *discretized features*. Table 4.2 shows the results of the animal labeling using equal width binning for different numbers of bins B. First, we use a fixed number of bins over every dimension. That

Figure 4.4: Some sample outputs from our system. 'GT' indicates ground truth, 'Predicted' indicates the predictions of the system.

is, along every dimension in the feature vector, the number of bins is set to a constant B. Note that irrespective of the number of bins, the performance has improved significantly. The precision has more than doubled, and the recall has improved by more than 20% in absolute value. *The discretization has improved the classification.* These findings are consistent with those of Dougherty et al. [20]. Overall, we see that these results are significantly better than all the baselines in Table 4.1. In addition to the discretization, the key aspect of this method using a naive Bayes classifier is that each component of the CNN representation is treated separately rather than using the full feature vector in a high-dimensional space. These bring robustness to image clutter and changing backgrounds, and help in learning from few examples.

Next, looking at the F_1 measures for different values of B, we see that the best results are obtained when $B = 7$. In addition to fixing the number of bins along every dimension, we used a heuristic to set a variable number of bins for each dimension. Using the heuristic proposed in the S-Plus histogram algorithm of Spector [92], we set the number of bins along each dimension to $log_2 l$, where l is the number of unique values in that dimension. Using this heuristic, different dimensions had different numbers of bins. We observed that of the 128 dimensions, 12 had 7 bins, while the rest had 8 bins. This explains why we have the best results in the range $B = 7$ and $B = 8$.

Figure 4.5: Some sample images from the Snapshot Serengeti [94] dataset, together with the descriptions that show the difficulty of the task. The green box indicates that the animal was recognized correctly, while the red box indicates that the animal was missed.

Table 4.3 shows the results of the animal labeling using equal frequency binning for different numbers of bins B. Here, since we are dealing with sparse matrices, we have to ensure that all zero-valued entries along a dimension should belong to the same bin. The results in table 4.3 incorporate this correction. As with the equal-width case, we obtain significant improvements over the naive Bayes classifier with continuous features.

Fig. 4.4 shows some of the sample outputs of our system. Note that our method is capable of identifying multiple species in the same frame, as well as detecting frames that do not contain any animal.

4.6.2 Transfer to Camera-trap Images

The second aspect of the evaluation is to measure how well the representations and models transfer to external data from an entirely different setup. To evaluate this, we use the Snapshot Serengeti [94] dataset, which consists of camera-trap (remote, automatic cameras) images covering wildlife in the Savanna. We learn animal recognition models using the weak labels of our dataset and apply them to the Snapshot Serengeti [94] dataset. It is important to note that the pictures

Method	P	R	F_1
CNN-M-128 (1000 classes)	21.98	20.38	21.15
CNN-M-128 (filtered to 19 classes of our dataset)	91.75	20.38	33.35
CNN-M-128 (filtered to 3 overlapping classes)	100	20.38	33.86
SVM continuous (on our 19 classes) - using weak labels	58.16	14.96	23.80
SVM continuous (on 3 overlapping classes) - using weak labels	86.34	14.96	25.50
SVM continuous (on 3 overlapping classes) - using GT	100	9.31	17.04
NBC continuous (on 3 overlapping classes) - using weak labels	49.03	90.53	63.61
NBC continuous (on 3 overlapping classes) - using GT	62.07	67.71	64.77
NBC discretized into $log_2 l$ bins (on 3 classes) - using weak labels	53.45	65.73	58.95

Table 4.4: Performance of the animal recognition models learned using our data, applied on images from the Snapshot Serengeti [94] dataset.

of this Serengeti dataset are captured automatically, in very different scenes, under various illumination conditions. This causes a huge *domain shift*. The Serengeti dataset covers 40 mammalian species, of which three (Lion, Zebra and Hippopotamus) also appear in our video documentary dataset. We choose 500 random images[6] each of Lion and Zebra, and all 37 images available for the Hippopotamus class. This set forms the target data on which the animal recognition models will be tested. Fig. 4.5 shows some of the sample images from the Serengeti dataset.

Table 4.4 shows the performance of the animal recognition models learned using our data, applied on the target dataset. The first baseline is simply based on the probabilities output by the CNN pre-trained on ImageNet. We used the same architecture (CNN-M-128) that was used for feature extraction. When the output probability for a certain class was >0.5, we concluded that the system predicted that class. Of course, multiple classes could be predicted for each key frame. Although some of the classes predicted covered 'lake side', 'hay' etc. which were not explicitly labeled in our setup, there were a lot of animals incorrectly predicted (which did not belong to our dataset of 19 animals). These included elephant, panther, camel, dugong. We filtered the outputs to just retain the 19 classes that were seen in our dataset. This increased the precision by a large margin (second row in the table). Next, we retained only the three classes that were common to our dataset and the Serengeti dataset. While this gave a perfect precision, the recall stands low at approx. 20% in all the three cases above.

Next, we train an SVM (on the continuous features) on all the 19 classes of our dataset, using the weak association of the subtitles and applied them to the Serengeti [94] dataset (Second block on table 4.4). Note that the performance is

[6]shot between 6:00 am and 6:00 pm

low compared to the methods based on ImageNet (in the first block). *The model learned by the SVM on our dataset does not compare well with that of ImageNet, which was trained on several thousands of zebras, hippos and lions.* As with the previous block, filtering to the 3 relevant classes increases the precision by a large margin, while the recall stays the same. When we used the ground truth labels instead of the weak labels (which basically indicate if a frame could have some animal), we have a perfect precision, but the recall is even lower. By capturing elements in the background/environment which might be related to the animal, (e.g., a water body for the hippopotamus, or grasslands for the zebra), the training based on weak labels yields higher recall, albeit at the cost of precision.

The last block shows the performance using a naive Bayes, trained using both weak labels, and the ground truth. Again, we note that the precision is better with groundtruth labels, while the recall is lower. But in either case, there are remarkable improvements compared to the first and second blocks. *The idea of treating each component of the CNN representation individually rather than using the full feature vector in a high-dimensional space is crucial both for isolating the object(s) of interest from the clutter, and for learning with few examples.* The discretized naive Bayes does not perform better than the continuous naive Bayes in this case - the discretized features probably do not transfer as well to the target domain. Nevertheless, it certainly outperforms the classifiers in the first two blocks, by a large margin.

4.7 Conclusions

In this chapter, we investigate different image representations and models, including a support vector machine on top of activations of a pretrained convolutional neural network, as well as a naive Bayes framework on a *bag-of-activations*, where each element of the bag is considered separately. We show that the *bag-of-activations* perspective allows key components in the image to be isolated, in spite of largely varying backgrounds and image clutter, and eliminates the need for an object detection or image segmentation step. *In contrast to most work on integrated vision and language that use curated data, the proposed approach deals with vision and language that are complementary.*

When the source and target are of the same domain, we also found that the discretization used with a multinomial naive Bayes classifier yields much better performance compared to continuous features with a traditional naive Bayes classifier - the precision is more than doubled and the recall is boosted by more than 20% absolute for the task of identifying animals on a challenging dataset

of wildlife documentaries. Here, we have used unsupervised equal-width and equal-frequency binning of the features. In the future, we wish to explore other (weakly) supervised techniques for discretization, and their transfer to other domains.

In the next chapter, we use a structured predictor to leverage the interdependencies within and across the textual and visual modalities. The framework described next performs entity linking on text in addition to object recognition in the wild.

Chapter 5

Entity Linking across Vision and Language

We propose a novel weakly supervised framework that jointly tackles entity analysis tasks in vision and language. Given a video with subtitles, we jointly address the questions: a) What do the textual entity mentions refer to? and b) What/ who are in the video key frames? We use a Markov Random Field (MRF) to encode the dependencies within and across the two modalities. This MRF model incorporates beliefs using independent methods for the textual and visual entities. These beliefs are propagated across the modalities to jointly derive the entity labels. We apply the framework to a challenging dataset of wildlife documentaries with subtitles and show that this integrated modelling yields significantly better performance over text-based and vision-based approaches. We show that textual mentions that cannot be resolved using text-only methods are resolved correctly using our method. The work presented in this chapter is published as

Venkitasubramanian, A. N., Tuytelaars, T., and Moens, M.- F. Entity linking across vision and language. Multimedia Tools and Applications (2017) DOI:10.1007/s11042-017-4732-8

[. . .] this daybreak finds the kingfisher still digging. She$_1$ must be desperate. [. . .] A mink. I thought it was an otter when it burst out from the bank. One kingfisher had dived to safety, but which one? It was impossible to tell. The mink had been waiting in ambush, hidden, even from me, almost certainly attracted by the kingfishers' frantic whistling. She$_2$ stashed the first bird and returned, sure that there was another. But one kingfisher got lucky. She$_3$ spotted me.

Figure 5.1: An example of a subtitle excerpt together with the associated frames.

5.1 Introduction

It is estimated[1] that it would take an individual more than 5,000,000 years to watch the amount of video that will cross global Internet Protocol (IP) networks each month in 2020. Therefore, it is imperative that we have tools that will enable us to search and find not only relevant videos in a corpus, but also relevant snippets within a video. Towards this goal of making videos 'searchable', we consider a wildlife documentary with subtitles and address two problems that typically occur in such videos: a) Mapping the mentions in the subtitle to the correct animal name (Entity linking); and b) Identifying animals in the video key frames (Animal labeling).

These seemingly unrelated problems are quite closely coupled in reality. Particularly in videos, language and vision are complementary to each other and it is essential to look at them in unison. Vision tasks often benefit from the associated text [97] while Natural Language Processing (NLP) tasks benefit from the vision. As an example, consider Figure 5.1. Here she$_1$ refers to the kingfisher and she$_2$ refers to the mink. While there is no ambiguity in these two cases, resolving she$_3$ is not straight-forward. This piece of text might suggest that she$_3$ refers to the kingfisher, but in reality, it refers to the mink. The use

[1] http://www.cisco.com/c/dam/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.pdf (accessed May 15, 2017).

of the associated frames makes this clear. Especially in multi-modal settings such as ours, *the visual component is crucial for the correct resolution of the textual mentions*. In this work, we tackle the problem of entity linking and animal labeling in text and vision jointly.

In a weakly supervised setting, this problem presents a host of challenges for vision, text and the association of text and vision. On the vision side, we deal with a scenario where there are no visual demarcators to indicate the location of an animal. In fact, it is not even known if there are animals at all in a certain key frame. Additionally, since we are dealing with animals shot in their natural habitat, there are challenges due to self-occlusion, camouflage, illumination etc. On the textual side, while we have tools [21, 54] to detect entity mentions in the text, not all of them are pertinent to animals. Even when the mentions refer to animals, they are often so ambiguous (e.g. *'targets for the crocodile'* and *'the predators'*) that it is impossible to resolve them correctly without a holistic understanding of the context. As far as the linking of text and vision is concerned, the absence of bounding boxes in the visual data coupled with the presence of ambiguous mentions in text makes it harder to reliably tie together the entities in vision and language, that is, there are no ready examples to show the association in a limited, diverse dataset.

In order to address the animal labeling task in vision and the entity linking task in language, we build a Markov Random Field (MRF) using the textual and visual entities. The MRF models the dependencies that exist in language (among the various mentions), in vision (among the frames) and across language and vision (depicting connections between a mention and an animal shown). For the textual entities, we use the state-of-the-art coreference resolution system of Durrett and Klein [21] and for the visual entities, we use the model of Venkitasubramanian et al. [97](i.e., chapter 3) which predicts animal labels in vision on a frame-by-frame basis. Using these as starting points, we apply belief propagation to draw inferences on text and vision jointly. Here, we use a structured predictor that leverages the continuity aspect inherent in videos. Building on the approach of Venkitasubramanian et al. [97] (chapter 3), we not only improve the recognition in vision using text, but also address the problem of entity linking in text using vision.

The key contributions of this chapter are as follows:

1. We propose a novel probabilistic framework to jointly map the entities in vision and language, by capturing interdependencies within and across the modalities.

2. We propose a method to filter mentions and retain only those relevant for the context. In this work, we focus on the detection of mentions pertinent

to animals.

The rest of this chapter is organized as follows: section 5.2 discusses related work. Section 5.3 defines the problem. Section 5.4 describes our framework and Section 5.5 describes the detection of relevant mentions. In section 5.6, we provide the implementation details. Section 5.7 describes the experiments and results. Finally, section 5.8 provides the conclusions and future work.

5.2 Related Work

To the authors' knowledge, this is the first work that addresses the problem of entity linking across language and vision. Our task is at the confluence of a few other tasks: 1) Entity analysis tasks in text, 2) Animal labeling in vision, 3) Aligning text and vision, and 4) Cross-modal coreference resolution. In what follows, we describe the related work in each of these domains.

5.2.1 Entity Analysis Tasks in Text

The entity analysis stack in text comprises three tasks: named entity recognition, coreference resolution and entity linking.

While early work [15] formulates Named Entity Recognition (NER) as recognizing 'proper names' in general, the scope has since been widened to include certain 'natural kind terms like biological species and substances' [65]. Alfonseca and Manandhar [2] define NER as the task of finding and classifying objects that are of interest to us. The need for precise NER tools has lead to the development of several domain-specific approaches. For example, in the biomedical domain, several methods have been proposed to recognize gene or protein names, diseases, drugs etc. [56]. In our work, we focus on animals and identify the referents using a list of animal names from WordNet [63]. Additionally, we propose an approach to detect other mentions that are pertinent to animals using one of the most salient features used in NLP tasks, that is especially relevant to sentient beings - animacy. Furthermore, we propose a more generic method for detecting relevant mentions, that can deal with a wider class of objects that is not restricted to animals.

Coreference resolution is the task of finding all expressions that refer to the same entity in a text. The state-of-the-art probabilistic coreference resolution system is the model of Durrett and Klein [21]. The system is basically a Conditional Random Field (CRF) that takes in a set of 'surface-level' or general purpose

features together with a set of more sophisticated 'semantic' features (such as hypernymy, synonymy etc.). The state-of-the-art deterministic coreference resolver is that of Lee et al. [54]. This system relies on a set of rules applied one at a time in the order of decreasing precision.

The entity linking task focuses on mapping an entity to an entry in a knowledge-base. For an overview of various approaches to entity linking, we refer to [88]. The entity linking task is usually preceded by a Named Entity Recognition (NER) task. The classical entity linking task is already quite challenging due to name variations and entity ambiguity. In our setting, these challenges are far more pronounced. Resolving the mention *'targets for the crocodiles'* to *'zebra'* is far more difficult compared to resolving, for example, *'Cornell'* to *'Cornell university'* - the search space is much wider and the desirable outcome (*'zebra'*) is not apparent from the words in the mention.

Yet another interesting work is that of Durrett and Klein [22], where all the three tasks namely named entity recognition, coreference resolution and entity linking are tackled jointly. All the above approaches only apply to text. In contrast, we develop a method that exploits both textual and visual modalities to perform entity linking.

5.2.2 Animal Labeling in Vision

The problem of aligning animals from videos with their mentions in subtitles has been studied in [23] and [97]. The former approach relies on hand-annotated bounding boxes to localize the animals in a frame and uses an Expectation-Maximization (EM) algorithm to map the bounding boxes to the correct animal names. The latter learns classifiers from ImageNet [19] and adapts them to the target dataset using an EM algorithm. Our approach is different from these in a couple of ways. Firstly, both the approaches target only the vision side, while we address the problem on the language side as well. Secondly, even on the vision side, the approaches of [23] and [97] were applied only on a frame-by-frame basis, whereas our approach uses structured prediction that leverages the dependencies within frames.

5.2.3 Combining Text and Vision

Recently, there has also been some work on alignment across modalities for recognizing people [71, 70, 37]. These approaches rely on the use of a face-detector. While there are face detectors available with reasonable accuracy, there are no such detectors that allow localizing animals. As noted by

Venkitasubramanian et al. [97], the absence of the bounding boxes complicates the problem in many ways. Further, these approaches only use the names of people on the textual side, while we address a much broader problem of mapping any phrase that indicates an animal (e.g. *the neighbor*) to the right animal.

There has also been considerable interest in sentence/caption generation from images as well as natural language based object detection [44, 28, 36, 45]. These approaches are not directly applicable to our setting. First, we have too few data to train similar models. Second, in our context, the subtitles and the visuals are not parallel, but complementary. For example, often a few animals are mentioned in the text, while the connected frame only shows one of them. The connection between the vision and the text is therefore much weaker.

5.2.4 Cross-modal Coreference Resolution

Ramanathan et al. [76] address the problem of coreference resolution in a multimodal setting involving people in video sequences together with their names in the text. While this approach also handles ambiguous mentions such as *the man*, *the engineer* etc. on the textual side, on the visual side, it is used in a much cleaner setting with faces in bounding boxes.

Another work that addresses coreference resolution in text and vision is that of Chen et al. [50] where images of furniture (represented as 3D cuboids) are mapped with natural language descriptions of the room. Our problem is much more complex in several ways. Firstly, on the vision side, we are dealing with animals which come in various shapes and flexible bodies instead of 3D cuboids. Second, on the textual side, we deal with more complex expressions such as *the caravan of predators* which do not explicitly state what they are referring to. Finally, the subtitles in our context are not intended to describe the visual appearances, but to augment them with extra information.

5.3 Task Definition

The input to our system consists of a wildlife documentary with subtitles. On the textual side, we have a set of sentences in the subtitles. These sentences contain mentions $\mathbf{M} = \{m_1, m_2 \dots m_r\}$. Of these mentions, we are only interested in those pertinent to animals[2]. These mentions include the nominal mentions and the pronominal ones. From the nominal mentions, we can derive the set of

[2]We experiment with both gold mentions and those detected automatically using Section 5.5.

Crocodiles are everywhere. The migration is the bonanza **they**'ve been waiting for. But the zebra are surprisingly well-armed. Even in water, a zebra's kick is more than a crocodile can endure. As **each family** makes it safely to the other side, **its members** reassemble in a frenzy of greeting. But there are still plenty of **targets for the crocodiles**, who begin to step up **their** onslaught. Some **foals** [. . .]

they	→	crocodile
each family	→	zebra
its members	→	zebra
targets for the crocodiles	→	zebra
their	→	crocodile
foals	→	zebra

Figure 5.2: An example of the entity linking task in text. The unique names identified for this caption are crocodile and zebra (underlined), the mentions to be resolved are in bold.

unique animal names such as *penguin*, *lion* etc. denoted by $\mathbf{N} = \{n_1, n_2 \ldots n_p\}$. We do this by comparing each mention against a list of animal names derived from WordNet [63] as in [97].

On the visual side, we have key frames $\mathbf{F} = \{f_1, f_2 \ldots f_q\}$ which may or may not contain animals. Using the setup of [97], every frame is linked to five subtitles to the left and right of the frame. All mentions in this range of subtitles are also associated with the frame. Thus, we have a set $\mathbf{P}$ of mention-frame pairs $< m_i, f_j >$. Further, the *unique* names in this set of subtitles are also associated with the frame. Thus, with each frame f_j, we have a set of associated animal names $\mathbb{N}_j$.

Our objective is to jointly map the mentions $\mathbf{M}$ and frames $\mathbf{F}$ to the correct names $\mathbf{N}$. This is described by the two tasks below:

1. Animal labeling on the vision side ($\mathbf{F} \rightarrow \mathbf{N}$): The task is to identify what animals are in the frame. For each name $n_l \in \mathbb{N}_j$ corresponding to frame f_j, we have a binary variable y_l indicating presence or absence of animal n_l. The animal labeling task aims at identifying the most likely value of y_l corresponding to name n_l for frame f_j.

2. Entity linking on the text side ($\mathbf{M} \rightarrow \mathbf{N}$): Entity linking tries to associate each mention with a knowledge base entry [58]. In our case, this 'knowledge base' is the list of animal names derived from WordNet [63] as stated earlier. For each mention $m_i \in \mathbf{M}$, we have a set of possible names, which are the same as those corresponding to the frames associated with m_i. That is, for every mention m_i, we have a set of associated frames $\mathbb{F}_i$ derived from the set $\mathbf{P}$ of mention-frame pairs. Each frame $f_e \in \mathbb{F}_i$ has a set of associated animal names $\mathbb{N}_e$. Then, the names associated with the mention m_i are $\mathbb{N}_i = \bigcup_e \mathbb{N}_e$.

   The entity linking task aims at identifying the most likely name n_k from animal names $\mathbb{N}_i$ corresponding to mention m_i. Figure 5.2 shows an example of this task. Note that this includes mapping mentions such as *targets* and *victim* to the right animal, but we do not make the distinction between the different members of the same species. For example, *'the zebra that swam across the river'* and *'the one that watched him'* will both refer to the animal *Zebra*. This is what makes our task different from a classical coreference resolution task.

5.4 Our Approach

We use a probabilistic graphical model, specifically a Markov Random Field (MRF), to denote the relationships between the frames and mentions over the entire video. We have two kinds of nodes:

- Visual nodes $\mathbf{V}$: A frame may have several animals or none. To denote this, we use one node for every frame-animal name combination. This is a binary node that indicates the presence or absence of the animal in the frame. $\mathbf{V} = \{v =< f_j, n_k > | n_k \in \mathbb{N}_j$, the set of candidate names associated with $f_j\}$

- Textual nodes $\mathbf{T}$: As with the frames, we have one node for every mention-animal name pair to indicate whether a mention maps to a name. We

consider all the mentions that need to be resolved. $\mathbf{T} = \{t =< m_i, n_l > |n_l$ is associated with frame $f_e \in \mathbb{F}_i$ corresponding to $m_i\}$

In either case, the nodes are random variables and can have the value 1 or 0, indicating the presence or absence of a name for a given frame or mention.

Now, we build a bipartite graph $G =< \mathbf{V}, \mathbf{T}, E >$, with edges E across the visual and textual nodes, $\mathbf{V}$ and $\mathbf{T}$ respectively. The edges are built between any pair of nodes $v \in \mathbf{V}$ and $t \in \mathbf{T}$ iff frame f_j is associated with mention m_i (that is, $< m_i, f_j > \in \mathbf{P}$) and $n_l = n_k$. Figure 5.3 shows an example of the graphical model for a snippet from our video. This is an episode on zebra and crocodiles. Note that this graph has two connected components (blue and orange), one for each animal name. Also note that the graph has cycles (loops).

We use a Markov Random Field to infer the values of the hidden labels. For this global inference over the video, we experiment with two different paradigms: 1) Message-passing and 2) Particle-based or sampling methods. For the message passing, we use the sum-product or Loopy Belief Propagation (LBP) algorithm. Figure 5.4 illustrates the message passing using the cluster graph for one connected component (Zebra) of Figure 5.3. Note that every edge influences one textual and one visual node each and messages are passed from one cluster to another through the textual or visual nodes. The use of the bipartite graph ensures that beliefs are propagated among the vision nodes, through the associated textual nodes and vice versa, allowing the two modalities to influence each other. Pearl [69] showed that the belief propagation algorithm is exact if the graph is a tree and approximate when the graph contains cycles. Since our graph has cycles, the inference is approximate in our case. LBP has been used successfully both in computer vision [16] and natural language processing [82]. We also experiment with Viterbi (VIT) Approximation (or max-product algorithm) that is quite similar to LBP but works in the log space.

Yet another approach to inferencing is through sampling. We experiment with Gibbs sampling, in addition to LBP and VIT. In Gibbs sampling, we randomly sample instances from the distribution and use those as a sparse representation, which are used to re-estimate the probabilities iteratively, thus preventing the algorithm from getting stuck in a local maximum.

The potential functions ψ_{text}, ψ_{vision} and ψ_{text_vision} are initialized as follows:

- For each textual node $t =< m_i, n_l > \in \mathbf{T}$, the node potential $\psi_{text}(m_i, n_l)$ is obtained using the state-of-the-art probabilistic coreference resolver of Durrett and Klein [21].

  Their system estimates the probability that a certain mention is the back-pointer of some other mention, thereby generating a back-pointer

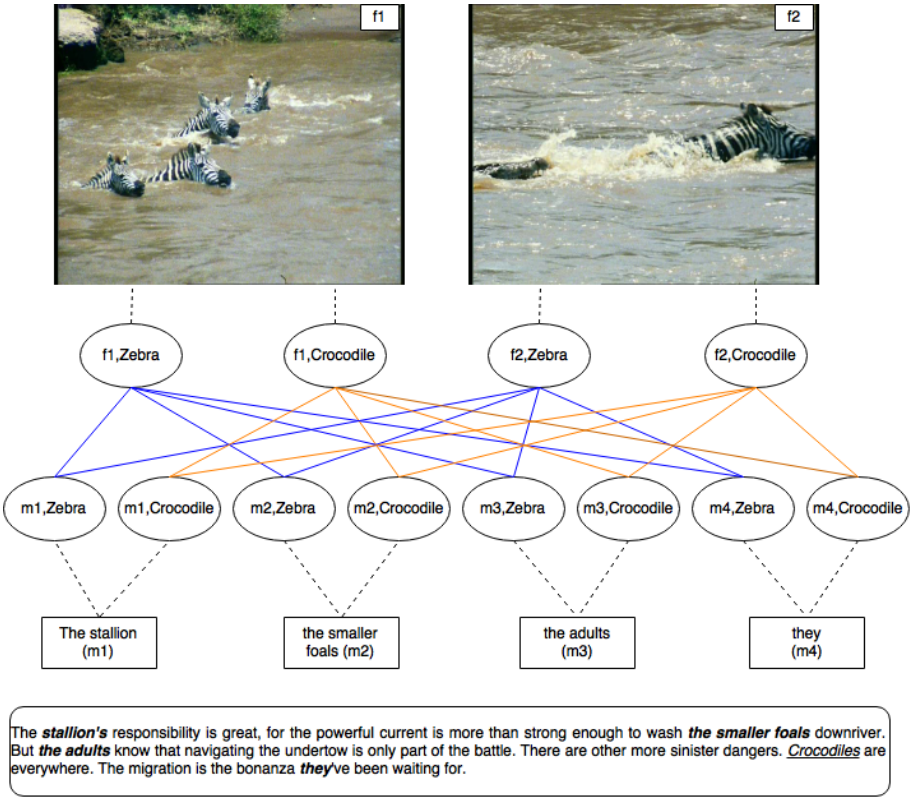


Figure 5.3: An example of a part of the graphical model built using two frames (first row) and the corresponding subtitles (last row). In the subtitles (last row), the mentions to be resolved are in bold and the unique names are underlined. The ovals denote the nodes in vision (second row) and text (third row). The blue edges denote the links among the zebra nodes, while the orange edges denote the links among the crocodile nodes. The mentions to be resolved (*The stallion*, *the smaller foals* etc.) are shown in the rectangles.

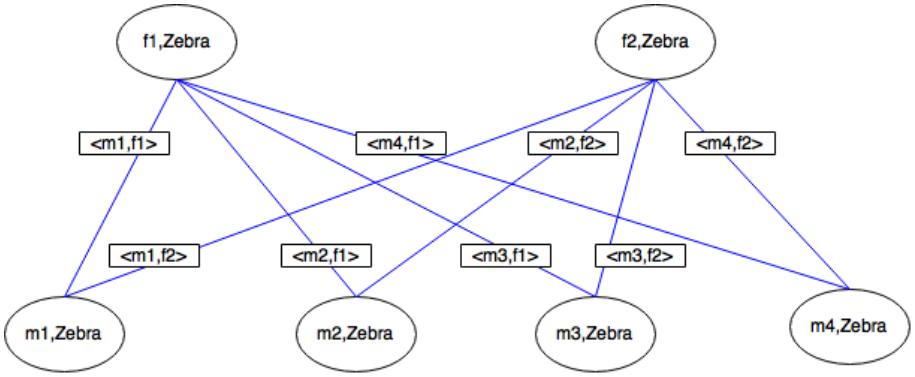


Figure 5.4: An example of the cluster graph for one connected component (Zebra) of Figure 5.3. The ovals denote the nodes in vision and text (first and last rows respectively). The blue edges denote the links among the zebra nodes.

probability matrix which will be used for our node potentials. We mirror the back-pointer probability matrix along the leading diagonal in order to have a symmetric matrix. Then, to obtain the node potential $\psi_{text}(m_i, n_l)$ corresponding to mention m_i and animal name n_l, we look at all the back-pointer probabilities associated with mention m_i, and filter them by those containing the name n_l. For example, suppose the mention to be resolved m_i is *'targets for the crocodiles'*, and we are interested in n_l = *Zebra*. Then, we look at the row in the back-pointer probability matrix corresponding to this mention and all the columns where there is a *'head_match'* with the name n_l. That is, we find all the columns where the head word is *'Zebra'*. Since there could be multiple mentions that contain *Zebra* as the head word, we choose the maximum[3] of these probabilities as the node potential ψ_{text}(*targets for the crocodiles,Zebra*).

$$\psi_{text}(m_i, n_l) = max_x\{p(m_i \rightarrow m_x)| \\ head\_match(m_x, n_l)\} \tag{5.1}$$

where $p(m_i \rightarrow m_x)$ denotes the back-pointer probability from mention m_i to mention m_x.

[3]We use maximum of the probabilities instead of mean or minimum because the most influential candidates are those that are closer to the said mention and have a high pair-wise score in the back-pointer probability matrix.

- For each visual node $v =< f_j, n_k >\in \mathbf{V}$, the node potential $\psi_{vision}(f_j, n_k)$ is defined as the strength of the connection between the frame f_j and name n_k. This is obtained from the probability $p(n_k|f_j)$ estimated using the model of [97] (Section 3.4 of chapter 3).
- The edge potentials $\psi_{text\_vision}(m_i, f_j)$ for the edges connecting the nodes are shared across the entire network and are obtained using a small validation set.

Using the above, the joint probability which will be maximized during the training of the network is given by

$$P(\mathbf{V}, \mathbf{T}) = \frac{1}{Z} \prod_{\substack{<m_i,f_j>\in\mathbf{P} \\ <m_i,n_l>\in\mathbf{T} \\ <f_j,n_l>\in\mathbf{V}}} \psi_{text}(m_i, n_l) * \psi_{vision}(f_j, n_l) * \psi_{text\_vision}(m_i, f_j) \tag{5.2}$$

where Z is the normalization constant.

Finally, to obtain the entity labels on the vision side, for each visual node $v =< f_j, n_k >\in \mathbf{V}$ we assign the label n_k to frame f_j if the node potential $\psi_{vision}(f_j, n_l) > 0.5$. This allows us to identify frames without any animal or with multiple animals of different species. On the textual side, since a mention typically refers to a single entity[4], we look at all the node potentials $\psi_{text}(m_i, n_l)$, corresponding to mention m_i and choose the name n_l that has the highest node potential.

5.5 Detecting Relevant Mentions

The text contains several mentions that refer to animals. These include nominal mentions such as *'penguin'*, *'the male'*, *'the mother'*, *'victims'*, *'cubs'* etc. and pronominal mentions such as *'she'*, *'he'*, *'they'* etc. To detect the mentions pertinent to animals, we experimented with two approaches. The first relies on the 'animacy' feature of a state-of-the-art deterministic coreference resolver, while the second uses a large database of hypernym relations extracted from the Web. These approaches are explained in detail below.

[4]It is possible that one mention actually refers to multiple animals. For example, *'The zebra and giraffe peacefully co-exist in Savannah.* ***They*** *are both very valuable to the wildlife ...'*. Here, the mention ***they*** refers to zebra *and* giraffe. However, we do not see such cases in our dataset, and ignore this case for simplicity.

5.5.1 Using the 'Animacy' Feature

The problem of detecting mentions that are relevant for animals largely boils down to finding whether a mention is animate or not. To find whether a mention is animate, we turn to a state-of-the-art deterministic coreference resolution system that uses 'animacy' as a feature. The coreference resolution system of Lee et al. [54] uses a set of sieves or rules, one at a time, ordered from highest to lowest precision[5]. One of these sieves relies on matching attributes such as animacy, gender and number of a mention with its antecedent(s). The 'animacy' feature which was built into the system turned out to be the second most important feature (next to number) for pronoun resolution. Their system sets one of the three values: ANIMATE, INANIMATE or UNKNOWN for the 'animacy' attribute. To detect relevant mentions, we first matched the mentions predicted by the probabilistic coreference system Durrett and Klein [21] with those predicted by the deterministic one of Lee et al. [54]. Then, we concluded that a mention corresponds to an animal if the mention is ANIMATE according to the system of Lee et al. [54] or if the head word contains an animal name. The reason we included the check for the head word is that we observed that several animal names such as *crocodiles*, *bear*, *dolphins* etc. were classified as INANIMATE. We also excluded first (such as I, me, myself, mine) and second person pronouns (such as you, your, yours) since they typically refer to the presenter and the viewer respectively. Since we are interested in animals in this work, the animacy feature is well-suited to this problem. However, in order to apply this work to other contexts involving inanimate objects, we propose an alternative approach.

5.5.2 Using a Hypernym Database

The second approach for detecting relevant mentions uses the WebIsADb database [87] containing more than 400 million hypernymy relations extracted from the CommonCrawl web corpus. This framework allows us to query by two attributes: class and instance, both of which have the sub-attributes: prefix, lemma and suffix. The output of the system is the frequency of that hypernym pair. For example, by specifying that the lemma of the class name is 'animal' and the lemma of the instance name 'penguin', we obtain a high frequency corresponding to this class-instance pair, indicating that 'penguin' may belong to the class 'animal'. Likewise, we query this database for the mentions identified earlier and find the score for it to be an animal. In particular, we query for

[5]This coreference resolution system is deterministic and does not provide the probabilities or strengths among mentions that is essential for our method. This prevents us from using this method from the start.

records in which the lemma of instance is the head word of the mention and lemma of the class is 'animal'. When the score obtained is greater than a certain threshold, we conclude that it is an animal. It is interesting that with this approach, the pronouns such as *he*, *she*, *him*, *her* etc. also had a high frequency for the animal class, indicating that these words are relevant for animals. Note that this approach is quite general and can be applied to any class of objects such as furniture, electronics etc.

5.6 Implementation Details

The pre-processing on the vision side is as described in Chapters 3 and 4. We used the CNN-M-128 architecture of [14], yielding 128 features.

The pre-processing on the text side comprises three steps: 1) identifying animal names; 2) detecting mentions; and 3) detecting the mentions pertinent to animals.

- To identify *animal names* such as *lion* and *zebra*, we use the entity detection step described in Chapter 3.
- To detect the *mentions* from the subtitles, we use the coreference resolution system of Durrett and Klein [21]. In particular, we used the Scala API of the Berkeley Coreference Resolution System[6].
- To refine these mentions to retain only those that are pertinent to animals, we adopted the approaches described in Section 5.5. We implemented two methods to detect relevant mentions: (i) using the 'animacy' feature and (ii) using the hypernym database.
  - Using the 'animacy' feature: We used the Stanford CoreNLP through their Java programming API, to detect mentions and extract their linguistic features, in particular the 'animacy' feature. For every mention detected earlier by the system of Durrett and Klein [21], we checked if there was a matching mention discovered through Stanford CoreNLP. When a match was found, we flagged it as relevant if the corresponding animacy value was ANIMATE. Additionally, if the head word of a mention corresponded to an animal name, we also flagged that as relevant. Finally, we excluded first and second person pronouns from the list of relevant mentions.

[6] http://nlp.cs.berkeley.edu/projects/coref.shtml (accessed on May 21, 2017).

– Using the hypernym database: WebIsADb hypernym database[7] provides two options for downloading the tuples: grouped by instances or by class names. We downloaded the tuples grouped by class name. Each tuple consits of an id, instance, class, frequency etc. We filtered these records to retain only the instance and frequency where the class name was 'animal'. With this list as reference, we scanned all mentions detected by Berkeley Coreference Resolution System, to check if the head word was mentioned as the lemma in any of the tuples. If it was present and the frequency > threshold, we flagged the mention as relevant.

To build the ground truth animal mentions for the evaluation, we manually refined the list of mentions predicted by the coreference resolution system of Durrett and Klein [21] and tagged them with the correct animal from the list of animals in WordNet [63].

For building the graphs and performing the inference, we used the MATLAB implementation of [86]. Code for detecting relevant mentions, building graphs and inferencing on them is available at https://github.com/aparnavenkit/Entity-Linking-across-Vision-and-Language.

5.7 Results

The dataset is the same as that of Chapters 3 and 4. There were 602 frames on the vision side. Figure 5.5 shows some sample images from our dataset. There are various challenges due to the nature of animals (flexible bodies, self-occluding, displaying large variation in pose) and due to the nature of the video set up (natural habitat accounts for camouflage and occlusion by environmental elements).

On the textual side, the subtitles contain 7,304 words in 545 sentences. There are 206 annotated mentions pertaining to animals, of which 89 are pronouns and 41 names of animals (including repetitions). There are 19 unique animal names in total. The graph built using the textual and visual nodes over the entire dataset has 826 nodes and 3585 edges.

The evaluation covers three aspects: 1) evaluation of mention detection; 2) evaluation of entity linking; and 3) evaluation of the animal labeling.

[7] http://webdatacommons.org/isadb/ (accessed on May 21, 2017).

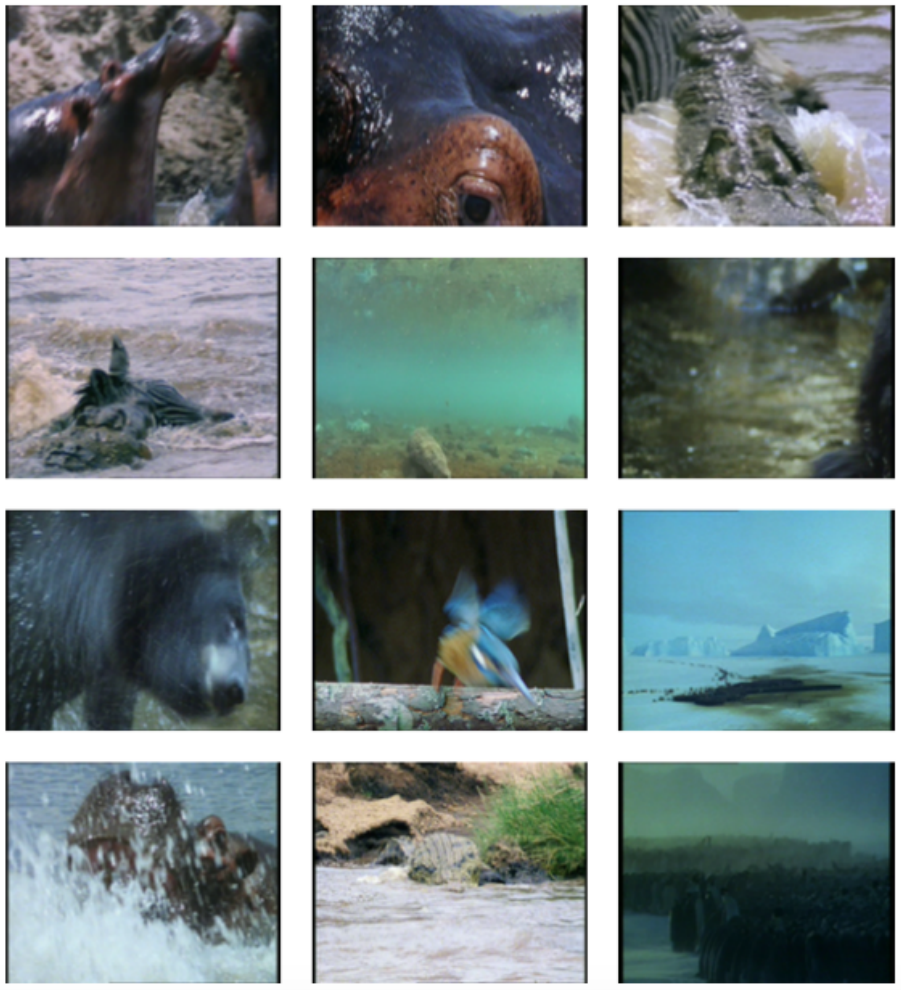


Figure 5.5: Challenges from the vision side: (i) Random poses where key distinguishing features of the animal are absent, (ii) Multiple species in same image, (iii) Blurry images, (iv) The animal to be recognized is blurred out, (v) The animal to be recognized is too far from the camera, (vi) Self occlusion, (vii) Poor illumination, (viii) Occlusion due to environment/ context and camouflage. All pictures depicted above have one or more of these issues.

Precision	Recall	F_1
91.05	84.8	87.81

Table 5.1: Results of the mention detection using the 'animacy' feature described in Section 5.5.

5.7.1 Detecting Relevant Mentions

Table 5.1 shows the results of the mention detection using the animacy feature of [54]. The performance is quite good in terms of both precision and recall. Recall that we also used WordNet's [63] animal list to ensure that noun phrases containing animal names such as *the mink*, *the penguins*, *a young white bear* were not classified as INANIMATE. Most of the errors were due to the absence of a perfect match between the mentions predicted by the two coreference resolution systems ([21] and [54]), often due to different segmentation of the candidate mentions. For instance, the total number of mentions predicted by the system of Durrett and Klein [21] was 560, whereas that by Lee et al. [54] was 526, including INANIMATEs and UNKNOWNs. As an example, consider the sentence below:

> *Each female times her return to coincide with the hatching of her chick.*
> Mention identified by system of Lee et al. [54]: *'Each female times her return to coincide with the hatching of her chick'*
> Mentions identified by system of Durrett and Klein [21]: *'Each female times her'* and *'the hatching of her chick'*

Some of the other misses were due to forms of 'it' being classified as INANIMATE when they actually referred to animals. For example, *'To get off the beach, the killer has to thrash its body'*. But, it doesn't make sense to override the animacy prediction of [54] and include all forms of 'it', since 'it' is often used as a syntactic expletive (for example, *It is raining*). Other misses include more general classes of animal names such as *mammals*, *cubs* etc. There were also some false positives, where an INANIMATE concept was classified as ANIMATE, based on the context. For example, *It's a way of fending off evil spirits.* There were a few instances where a mention is indeed animate, but does not refer to animal(s). For example, *The Ainu celebrate this special event with their own dance.* Both *The Ainu* and *their* were correctly classified as ANIMATE, but were not relevant for our problem. Apart from these rare cases, this approach worked quite well. It is interesting to note that some of the mentions like *the intruder*, *the homeless female*, *this bedraggled survivor*, *these males*, *the other youngster* were correctly identified as ANIMATE.

Threshold	Precision	Recall	F_1
1	39.55	95.58	55.95
12	43.41	92.15	59.02
100	56.25	88.23	68.70
150	60.62	85.29	70.87
190	63.46	84.31	72.42
195	63.56	83.82	72.30
200	**63.80**	**83.82**	**72.45**
250	60.85	70.09	65.14
300	61.13	68.62	64.66

Table 5.2: Results of the mention detection using the Hypernym Database WebIsADb [87].

Next, we evaluate the mention detection using the Hypernym Database WebIsADb [87]. Table 5.2 shows the results for different thresholds. The threshold is basically the frequency above which a pair can be considered as a hyponym-hypernym pair. It makes sense that increasing the threshold increases the precision and decreases the recall. It is interesting that in most cases (threshold ∼25 and above) mentions containing *victims*, *targets* etc. were also classified as ANIMATE which is useful in our context. However, there is also noise, for example, rivers seem to have a high association with animals in this database, but are nevertheless irrelevant for our problem. The best result (F_1 of 72.5%) obtained when the threshold is set at 200, is still ∼15% short of those of the previous approach. Nevertheless, this approach has the advantage that it could be applied to any class of objects, and is not restricted to sentient beings.

5.7.2 Entity Linking in Text

To evaluate the entity linking on text, we consider three scenarios: 1) using gold mentions, 2) using the mentions detected using the 'animacy' feature in Section 5.5.1, and 3) using the mentions detected using the hypernym database in Section 5.5.2. In either case, we experiment with two methods of initializing the vision nodes: 1) using binarized CNN features pre-trained on ImageNet as in [97]; and 2) using the output of [97].

Init is the baseline obtained by using the back-pointer probabilities from the coreference resolution system of Durrett and Klein [21][8]. This system uses *only*

[8]Here, we look at the initial set of node potentials (obtained using the back-pointer probabilities from the coreference resolver of Durrett and Klein [21]) and assign each mention to the name with largest probability.

the text, (that is the subtitles in our context) and does not incorporate any visual inputs to resolve coreferences. LBP refers to Loopy Belief Propagation. Gibbs and VIT refer to Gibbs sampling and Viterbi Approximation respectively. For evaluating the textual task, we use the standard metrics [74] for coreference resolution - MUC [98], B^3 [3], and $CEAF_e$ [60], as well as their average, the CoNLL metric, all computed from the reference implementation of the CoNLL scorer [73] (see appendix for an overview of these metrics). We evaluate our methods only on the mentions that are ambiguous, that is, we leave out the animal names such as *zebra*, *crocodile* etc.

Table 5.3 shows the results of the entity linking on all the gold mentions (nominal and pronominal). Gold mentions are the ground-truth (manually annotated) mentions pertinent to animals. In Table 5.3, comparing the two methods of initializing the vision nodes, we see that using the output of [97] gives a significantly better performance. This is because these outputs give a better indication of the presence or absence of an animal in a key frame, as shown in [97]. Thus, *better vision probabilities lead to better performance on the language task.*

We performed a qualitative analysis to understand how the vision helps to better disambiguate the text. We found that mentions such as *'the male'*, *'the female'* and *'the bird'* are used throughout the subtitles, but refer to different animals at different points in time. For example, *'the bird'* refers to a *penguin* at a certain point, and then refers to a *Japanese crane*, a few sentences later. The use of associated video key frames helps disambiguate ambiguous mentions such as these.

Comparing the outputs of the 'Init' system with ours, we found that some of the mistakes in the 'Init' system are due to difficulties arising out of spoken text. In the subtitles, there are no paragraph breaks. As a result, the end of a topic and the beginning of the next are not clear. For example, consider the subtitle snippet below.

> *The splash tetra must have the most labour-intensive childcare of any fish.*
> *But his eggs are safer from predators on leaves rather than in the river.*
> *After two days of hard splashing, the fry emerge.*
> *Within minutes, this nervous herd will fragment into hundreds of individual families, as each stallion attempts to shepherd his mares and foals across.*

In this example, the first three sentences are about the splash tetra, while the last is about the zebra. In spoken text such as subtitles, there are no markers

to indicate the change of topic.

Sentence structures may also be different in spoken text, which makes it difficult for most coreference systems trained on well-written documents (e.g., news articles), such as that of Durrett and Klein [21]. For example,

> *In the rivers and lakes of Africa lives an animal which has a reputation for being the most unpredictable and dangerous of all.*
> *Even crocodiles are wary.*
> *The hippopotamus.*
> *Supported by the water, they$_1$ use less energy than they would on land. Moving requires only a gentle push. They$_2$ save energy in other ways too.*

In the example above, both *they*$_1$ and *they*$_2$ refer to the hippopotamus; the system of Durrett and Klein [21] maps them to crocodile while our approach identifies them correctly. The use of the visuals, together with the joint learning clearly overcomes these issues.

Second, comparing the different algorithms LBP, Gibbs and VIT, their performances are somewhat similar, although LBP is the best when using the output of [97]. In any case, using any algorithm, any mode of vision initialization, the methods outperform the initialization (Init) based on the coreference resolution in text [21] by a significant margin. LBP with an average of 83.2%, initialized with [97] has a gain of over 4% compared to the baseline at 79.2%[9].

Table 5.4 shows the results of the entity linking on all the pronominal gold mentions. The findings here are consistent with those of Table 5.3. Comparing these results with Table 5.3, note that the Average F_1 is better than those of Table 5.3. In general, resolution of pronouns such as *he*, *she*, *their* etc. is easier than resolution of mentions such as *targets for the crocodiles*, *the victim* or *this bedraggled survivor*.

Table 5.5 shows the results of the entity linking on all the mentions (nominal and pronominal) detected using the animacy feature. Again, we have the same trend as Tables 5.3 and 5.4. There is a significant increase in average F_1 compared to the baseline that uses the text-only coreference resolution of Durrett and Klein [21]. Also, *better initialization of vision nodes leads to better performance on text*. Note that these results are in general lower than those reported in Tables 5.3 and 5.4. The reason is that here we have used the mentions detected automatically using Section 5.5.1, instead of using the gold mentions. So,

[9]We tested the statistical significance of the results using a mention-level paired t-test and found that the LBP method was significantly better than Init ($p < 0.01$)

Method	B^3	MUC			$CEAF_e$	Avg
	F_1	P	R	F_1	F_1	F_1
Init	94.3	69.91	69.23	69.57	73.77	79.213
Initialization of Vision nodes using ImageNet [19] as in [97] (Chapter 3, section 3.5.2)						
LBP	92.74	73.55	73.43	73.49	75.00	80.41
Gibbs	94.3	76.18	73.64	74.89	74.8	81.33
VIT	94.3	75.48	72.27	73.84	76.43	81.52
Initialization of Vision nodes using the output of [97] (Chapter 3, section 3.4.4)						
LBP	94.81	77.03	76.8	76.91	78.01	**83.24**
Gibbs	94.81	77.10	75.32	76.20	76.33	82.45
VIT	94.81	77.49	74.26	75.84	78.01	82.89

Table 5.3: Results of the entity linking task using all gold mentions - nominal and pronominal.

Method	B^3	MUC			$CEAF_e$	Avg
	F_1	P	R	F_1	F_1	F_1
Init	95.33	85.52	77.61	81.37	74.49	83.73
Initialization of Vision nodes using ImageNet [19] as in [97] (Chapter 3, section 3.5.2)						
LBP	95.33	85.52	78.18	81.69	75.59	84.20
Gibbs/VIT	95.33	86.11	79.46	82.65	77.41	**85.13**
Initialization of Vision nodes using the output of [97] (Chapter 3, section 3.4.4)						
LBP	95.33	85.72	78.8	82.11	76.55	84.66
Gibbs/VIT	95.33	86.11	79.46	82.65	77.41	**85.13**

Table 5.4: Results of the entity linking task using gold pronouns.

the errors due to mention detection are also included. Nevertheless, there are significant improvements over the baseline, with 4% improvement in F_1 measure, while using LBP with initialization based on [97].[10]

Table 5.6 shows the results of the entity linking on all the mentions (nominal and pronominal) detected using the hypernym database, with the threshold set to 200 based on the results in Table 5.2. As before, we note the significant increase in average F_1 compared to the baseline that uses the text-only coreference

[10]The global inferencing over text and vision for the entire video was quite fast. On an Intel Xeon CPU E5-2687W processor with 3.10GHz, the LBP took 0.65997 seconds, while VIT and Gibbs took 0.76278 and 0.67559 seconds respectively.

Method	B^3			MUC			$CEAF_e$			Avg
	P	R	F_1	P	R	F_1	P	R	F_1	F_1
Init	78.23	84.83	81.4	50.98	57.46	54.03	57.39	52.61	54.9	63.44
Initialization of Vision nodes using ImageNet [19] as in [97] (Chapter 3, section 3.5.2)										
LBP	77.2	83.7	80.32	52.69	61.08	56.58	57.91	53.09	55.39	64.10
Gibbs	79.27	85.95	82.47	57.72	65.82	61.5	61.61	56.47	58.93	67.63
VIT	78.75	85.39	81.94	56	62.33	58.99	60.37	55.34	57.74	66.22
Initialization of Vision nodes using the output of [97] (Chapter 3, section 3.4.4)										
LBP	79.27	85.95	82.47	57.92	64.24	60.92	61.94	56.78	59.25	**67.55**
Gibbs	79.27	85.95	82.47	58.36	64.33	61.2	61.17	56.07	58.51	67.39
VIT	79.27	85.95	82.47	56.17	63.75	59.72	59.98	54.98	57.38	66.52

Table 5.5: Results of the entity linking on all mentions (nominal and pronominal) detected using the animacy feature of [54].

Method	B^3			MUC			$CEAF_e$			Avg
	P	R	F_1	P	R	F_1	P	R	F_1	F_1
Init	78.75	59.37	67.70	56.4	33.16	41.77	44.99	44.99	44.99	51.49
Initialization of Vision nodes using ImageNet [19] as in [97] (Chapter 3, section 3.5.2)										
LBP	78.75	59.37	67.70	60.58	34.93	44.31	48.40	48.40	48.40	53.47
Gibbs	78.75	59.37	67.70	59.52	34.28	43.51	47.06	47.06	47.06	52.76
VIT	78.75	59.37	67.70	56.62	33.61	42.19	45.80	45.80	45.80	51.90
Initialization of Vision nodes using the output of [97] (Chapter 3, section 3.4.4)										
LBP	79.27	59.76	68.15	62.99	35.25	45.2	48.22	48.22	48.22	**53.86**
Gibbs	78.75	59.37	67.7	61.21	35.23	44.72	48.60	48.60	48.60	53.67
VIT	79.27	59.76	68.15	58.00	34.66	43.39	47.49	47.49	47.49	53.01

Table 5.6: Results of the entity linking on all mentions (nominal and pronominal) detected using the Hypernym Database WebIsADb [87].

resolution of Durrett and Klein [21]. These results are also lower than those based on the animacy based approach (reported in Table 5.5) by about 14%. Recall that the F_1 of the hypernym based mention detection is ∼15% lower than that of the animacy based approach.

Figure 5.6 shows some sample entity labels generated by our system using gold mentions and LBP. Note that the system correctly resolves the ambiguous nominals such as *a big land animal*, *some foals* etc. Some of the errors in our system occur when there were multiple animals in a frame, and neither the back-pointer probabilities [21] in text nor the vision probabilities from [97] gave a good estimate of the probability of the animal name given the frame/mention

to start with. One such example is the misclassification of *'many'* to crocodiles when they actually referred to zebras (Figure 5.6). The last text excerpt in this figure is really interesting. All the pronouns in this example actually refer to mink, but just looking at the text leads us to believe that they refer to the *kingfisher. It is impossible to resolve these mentions correctly without the vision.*

5.7.3 Animal Labeling on Vision

As with the entity liking on text, we experiment with three scenarios: 1) Gold mentions, 2) mentions identified using the animacy feature and 3) mentions identified using the hypernym database. Table 5.7 shows the results of the animal labeling on vision for all the scenarios. The baseline (Init) is the initialization obtained using the output probabilities of Venkitasubramanian et al. [97]. Recall that the approach of [97] learns classifiers from ImageNet [19] and iteratively adapts them to the target dataset using textual cues from subtitles, particularly whether or not a certain animal name is mentioned in the subtitles corresponding to each frame in the target dataset. Their system outputs a set of probabilities $p(n_k|f_j)$ that any frame f_j contains animal n_k. The baseline (Init) that we used assumes that an animal n_k is present in frame f_j if the corresponding probability $p(n_k|f_j) > 0.5$.

While in the setting of [97] there was ambiguity due to the lack of reliable connections between the animal and name pairs across text and vision, here, we deal with even more ambiguity, since we have mentions such as *'the female'*. Despite that, we have significantly better performance with both gold and detected mentions. This is because improving the coreference resolution leads to more confident animal image-name pairs.

The performance of the three methods (LBP, VIT and Gibbs) are comparable, although LBP gives the best performance with a gain of close to 4% in F_1 compared to the Init baseline[11].

Further, when comparing the performance of the system using gold mentions with that using detected mentions, we note that the former is clearly better. This makes sense because false positive mentions (e.g. *'evil spirits'* and *'Ainu'* were classified as ANIMATE) will still be mapped to animal names on text and their probabilities will impact the rest of the graph. *Better text leads to better results on the vision side.*

[11] We tested the statistical significance of the results using a frame-level paired t-test and found that the LBP method was significantly better than Init both in terms of precision ($p < 0.001$) and recall ($p = 0.0093$).

In the rivers and lakes of Africa lives an animal which has a reputation for being the most unpredictable and dangerous of all$_1$. Even **crocodiles** are wary. The **hippopotamus**. Supported by the water, they$_2$ use less energy than they would on land.	1. an animal which has a reputation for being the most unpredictable and dangerous of all GT: hippopotamus Predicted: hippopotamus 2. they GT: hippopotamus Predicted: hippopotamus
There's more than one **young bear** around, and some lessons have to be learned by way of a mistake. But for this youngster$_1$, it's worth checking to see if the other youngster$_2$ knows something he$_3$ doesn't. But no. A truth is confirmed. In the water, a big land animal$_4$ is no match for quick slippery fish$_5$.	1. this youngster GT: bear Predicted: bear 2. the other youngster GT: bear Predicted: bear 3. he GT: bear Predicted: bear 4. a big land animal GT: bear Predicted: bear 5. quick slippery fish GT: salmon Predicted: salmon
Some foals$_1$ are strong enough to defend themselves$_2$ in spite of the **crocodiles**' determination, and many$_3$ succeed in repelling their$_4$ attackers$_5$.	1. Some foals GT: zebra Predicted: zebra 2. themselves GT: zebra Predicted: zebra 3. many GT: zebra Predicted: crocodile 4. their GT: zebra Predicted: zebra 5. their attackers GT: crocodile Predicted: crocodile
But **one kingfisher** got lucky. She$_1$ spotted me. We were so absorbed in the fight that she's$_2$ as surprised to see me as I was to see her$_3$.	1. She GT: mink Predicted: mink 2. She GT: mink Predicted: mink 3. her GT: mink Predicted: mink

Figure 5.6: Some sample outputs from our system using gold mentions and LBP. The left column shows the subtitle text. Bolded mentions contain an animal name, while the underlined mentions are the ones to be resolved. The right column shows the outputs. 'GT' indicates ground truth, 'Predicted' indicates the predictions of the system.

Another interesting aspect of the evaluation regards the performance of the algorithm when more than one name is associated with a frame. In our dataset, there are 65 frames with more than one animal shown, and 153 frames with more than one possible name assigned as weak labels (using the nearby subtitles). Using gold mentions with the LBP algorithm, we obtained a precision of 82.46% and recall of 83.93% (yielding an F_1 of 83.19%) in the first case, and a precision of 59.14% and recall of 91.67% (yielding an F_1 of 71.90 %) in the second case. The latter case is harder, making the F_1 slightly lower than that on the entire dataset (73.42%). While these cases are certainly challenging, even identifying frames with just one animal or none is not straightforward, since it is not known in advance whether or not there is an animal in the frame.

Yet another aspect of the evaluation is the impact of joint modelling. To measure this, we experiment with the system of Venkitasubramanian et al. [97] (Chapter 3) using the model proposed here to resolve entity mentions. Recall that the system of Venkitasubramanian et al. [97] uses a combination of coreference resolvers to resolve pronouns. The outputs of all the coreference systems were fed into an EM algorithm, which then used these as evidence to connect an animal image-name pair. We ran this system using our entity labels, instead of the combination of coreference resolvers. In this case, we obtained an F_1 of 65.59% (second row in table 5.7) for the animal labeling task. These results are lower compared to those obtained in [97], which was based on multiple coreference resolvers. This is because, in our setup, we consider more complex mentions including nominals which are ambiguous, instead of just using pronouns (which are easier to resolve compared to the ambiguous nominals we have in our dataset) and animal names. Some of the errors we obtained in the entity linking propagated through the system of [97], resulting in the lower performance. Comparing the approach that we propose in this chapter with the method of [97], we see a significant increase in the performance of the animal labeling, which is largely attributed to the joint modelling. *The improvement on the vision side is not just because of the better entity labels, but also due to the joint modelling.*

Figure 5.7 shows some outputs of our system with LBP. Our system performs quite well despite various challenges such as blurry subject, random poses etc. The system is capable of identifying multiple species in the same frame, as well as detecting frames that do not contain any animal.

Figure 5.7: Some sample outputs from our system using LBP. 'GT' indicates ground truth, 'Predicted' indicates the predictions of the system.

Method	Precision	Recall	F_1
Init	57.27	88.73	69.61
Approach of [97] using our entity labels	53.22	85.45	65.59
Using gold mentions			
LBP	65.99	82.74	**73.42**
Gibbs	61.84	88.83	72.92
VIT	62.83	85.79	72.53
Using mentions detected based on the animacy feature			
LBP	58.18	89.72	**70.59**
Gibbs	58.23	89.25	70.48
VIT	57.52	91.12	70.52
Using mentions detected based on the hypernym database			
LBP	59.30	86.45	**70.34**
Gibbs	59.42	85.51	70.12
VIT	57.31	89.72	69.95

Table 5.7: Results of the animal labeling task on the visual data using gold mentions and mentions detected automatically.

5.8 Conclusions

This chapter shows that the joint modelling of entity linking tasks in vision and language results in better performance in both modalities. The framework proposed incorporates beliefs from state-of-the-art methods independently from text and vision, and performs global inference over the whole video using a structured predictor. We have shown that the performance of the entity linking has improved through the use of visual cues while that of the animal labeling has improved through the use of better textual coreference resolution. Furthermore, we have demonstrated the use of our method for textual mentions that cannot be resolved using text-only methods. While the framework has been validated on a wildlife documentary here, the methods proposed are quite generic and can be applied to various other scenarios involving language and vision, such as aligning people's names with faces or furniture and other objects.

In the future, we wish to apply these methods to other datasets involving a wide variety of subjects. Further, we would like to extract the interesting regions in the picture which can contribute to a better performance. The approaches described advance us towards automatic multimedia indexing.

Chapter 6

Conclusions

With the proliferation of multimedia data on the web, tools to accurately and efficiently understand and index videos are more important than ever before. We break down the video understanding task into two components - *object recognition in-the-wild* and *entity linking* on the text. In this thesis, we propose methods to address these tasks, and tackle the challenges that occur in a realistic video setting, as opposed to using clean, curated data with carefully assembled training examples.

6.1 Thesis Summary and Highlights

In Chapter 3 we start out by investigating image representations and object recognition models demonstrated on the task of wildlife recognition. This chapter addresses the first set of research questions:

Can we build object recognition models that can deal with a noisy, *'in-the-wild'* setting, as opposed to using clean, curated data, with carefully annotated labels? Can these models work with subtitles that are complementary to the vision, in lieu of transcripts or textual descriptions that are more parallel and provide a complete, accurate account of what is shown in the images or the video? Can we leverage external labeled datasets to learn object recognition models to overcome the lack of sufficient, reliable training data? How can these models be adapted to a multi-modal context involving vision and language?

In this chapter, we proposed a weakly supervised framework that learns a model from an external labeled dataset (ImageNet) and iteratively adapts it to the target dataset based on textual cues from the subtitles. In particular, we build on two interesting properties [57] of CNN activations: 1) the features preserve their essence even after binarization and 2) they can be treated independently along the dimensions. Based on the first property, we represent an image with binarized CNN activations, and think of them as indicating the presence or absence of some aspect of the image. This is an intuitively appealing representation - using this representation, we can measure how the presence (or absence) of an animal label contributes to the presence (or absence) of a visual feature. This is measured by the probability of the feature given the animal name, initially using an external labeled dataset (ImageNet). Further, the independence property of the CNN features allows us to combine the probabilities of different features for the animal name in a naive Bayes construction to obtain the likelihood of the name for the frame. In turn, the likelihoods of the names for the frame can be used to re-estimate the probabilities of different features for the animal name, effectively adapting to the target data. The process continues until convergence.

We find that although ImageNet contains several thousands of labeled images for each class, the models learned from ImageNet applied 'as is' to our dataset do not perform well; it is beneficial to adapt these to the target dataset. We show that *by training classifiers on an external labeled dataset, and adapting them iteratively to the target dataset, using textual cues, the accuracy of classification can be improved*. In particular, *the accuracy of our approach is significantly better than a) a purely vision-based approach or b) purely text-based approach or c) an approach that uses both text and vision, but without labeled examples or d) an approach that uses both text and vision, and labeled out-of-domain examples (from ImageNet), but without the adaptive learning.*

Next, we further addressed some of the challenges above, without the use of external training data. Chapter 4 is based on the following research questions:

Can we build image representations and object recognition models that deal with the challenges above (noisy, *'in-the-wild'* setting, and complementary data), while also coping with the lack of external training data? How well do these models transfer to unseen images of an entirely different domain, captured across diverse topographical regions under vastly varying environmental conditions and illumination settings?

In this chapter, we consider two perspectives on the feature vector of CNN activations: (i) *a global perspective* where each feature vector is treated as one entity, and (ii) a *'bag-of-activations' perspective*, where each element of

the bag is considered separately. The global perspective is by far the most commonly used and fits well with a linear Support Vector Machine (SVM) classifier. In contrast, the *'bag-of-activations'* paradigm introduced here is used in conjunction with a naive Bayes framework and allows to treat each element of the CNN representation individually rather than using the full feature vector in a high-dimensional space. This has two benefits. Firstly, by considering individual aspects or components of the images, *this representation brings robustness to image clutter and changing backgrounds*; this means the representation and model can be used 'in-the-wild' setting, without necessitating bounding boxes that localize objects of interest (e.g., animals in our case). The second benefit is that by working on a per-dimension basis, this model and representation can effectively *deal with the lack of reliable training data.* It is worthwhile to note that the CNN activations used here are shown to have the property that they can be treated independently along the dimensions.

Another interesting aspect is that the image representations and object recognition models present in this chapter are learned by simply watching a documentary. That is, we used the video key frames and the weakly associated subtitles to learn animal recognition models. In addition to testing the models on the source video documentary, we evaluated how well these models and representations transferred to an external dataset of a vastly different domain, involving a large domain shift. After an extensive evaluation of several models and architectures, we found that the *'bag-of-activations'* based model performed best not only on the source dataset, but also on the new, unseen domain.

Next, we turn our attention to exploiting the relationships within and across the visual and textual modalities in videos. Chapter 5 is dedicated to the following group of research questions:

In a multi-modal setting such as videos with subtitles, is there a model/representation that can encode dependencies within and across the modalities? Can we capitalize on the inherent characteristics of video documentaries (such as temporal continuity and the said interdependence) to build models that can jointly recognize the content of the video key frames and resolve the textual mentions in the subtitles? Can we automatically detect textual mentions that are relevant for the context (e.g., mentions relevant to animals, or those relevant to electronic equipment)?

Chapter 5 proposes a novel weakly supervised framework that jointly tackles entity analysis tasks in vision and language. Given a video with subtitles, we jointly address the questions: a) What do the textual entity mentions refer to? and b) What/ who are in the video key frames? We use a Markov Random Field

(MRF) to encode the dependencies among video key frames (e.g., temporal continuity), among the various textual mentions, and across the visual and textual modalities (depicting connections between a mention and an animal shown). This MRF model incorporates beliefs using independent methods for the textual and visual entities. These beliefs are propagated across the modalities to jointly derive the entity labels, as a structured prediction.

To cope with the numerous mentions irrelevant to a context (e.g, irrelevant for animal recognition), we propose two methods to automatically detect which mentions are pertinent. The first method uses one of the most salient features used in NLP tasks, that is especially relevant to sentient beings - animacy. The second method, on the other hand, is more generic and can deal with a wider class of objects that is not restricted to animals.

The techniques for mention detection, entity linking on text and animal recognition on vision are all evaluated on a challenging wildlife dataset. We have shown that *the joint modelling of entity linking tasks in vision and language results in better performance in both modalities.* We have proved that *the performance of the entity linking has improved through the use of visual cues while that of the animal labeling has improved through the use of better textual coreference resolution.* Furthermore, we have demonstrated the use of our method for *resolving textual mentions that cannot be resolved using text-only methods.*

To summarize our contributions, we identify three major themes:

- **On the vision side**: We have proposed weakly supervised methods to learn object recognition models. These models have been validated on a challenging dataset of wildlife documentaries, in a realistic 'in-the-wild' setting, without necessitating an object detection step.
- **On the language side**: We have built a full-fledged entity analysis system, spanning the entire pipeline from named entity recognition through coreference resolution and entity linking. Rather than subsisting with the conventional methods involving persons and organizations, we have broadened the scope to cover a wider variety of entities in a wildlife documentary, and demonstrated how the methods could be tailored to any domain of interest.
- **On the integration of vision and language**: We have proposed a framework that jointly understands the text and vision by capturing interdependencies within and across the modalities. All the methods explored in this dissertation have been applied to videos with weakly associated subtitles.

[. . .] A few days later, I found Maachli being pursued by Nick again. He's definitely interested in mating. He's much bigger than Maachli, so this is a really risky moment for her. Will she give in and let him mate with her or will she try and fight him off? It looks like she's keeping down, away from Nick, so he can't get behind her and mount her. Nick's really enticed by her scent now. There's a real tension in the air. Nick doesn't look like giving up. They're about to go for each other. Maachli still has the fight in her and it looks like she's managed to wound more than Nick's pride. This gash is going to make it hard for Nick to hunt over the next few days. [. . .]

Figure 6.1: A set of frames and subtitle excerpt from our dataset showing two tigers. Together with other vision based fine-grained categorization/biometric systems, our multimodal framework can be used to further improve the performance.

6.2 Future Work

Although the methods proposed in this thesis have been validated on wildlife datasets, they are quite generic and can be applied to a plethora of problems. We identify some interesting directions for future work below:

- **Fine-grained categorization and animal biometrics (individual recognition)**: Lately there has been considerable interest in individual recognition purely based on vision, for example, for identification of whale sharks [41], apes [24, 59], penguins [13] and various birds [91].

  Coupled with recent advances (e.g., [100]) in purely vision based fine-grained categorization, the joint modeling framework of Chapter 5 could be tailored to improve fine-grained categorization and animal biometrics, if there are narratives mentioning names of the individual animals. Figure 6.1 shows an episode from our dataset to illustrate this.

```
00:01:19--> 00:01:21
and the challenges she wanted to contain
00:01:22 --> 00:01:24
the IOM flyovers poll like gold can
00:01:25 --> 00:01:27
come back and AT&T are that if one of the
00:01:28 --> 00:01:30
tallest order thing anyone could ever get
near
00:01:48 --> 00:01:50
on Thursday the 26th of April 19
00:01:51 --> 00:01:53
23 a young Scottish aristocrat
00:01:54 --> 00:01:56
Lady Elizabeth Bowes Lyon later the
Queen mother
00:01:57 --> 00:01:59
married King George the fifth second from
00:02:00 --> 00:02:02
and from
00:02:04 --> 00:02:06
that day was captured on Tuesday News Real
```

```
00:01:18 --> 00:01:21
...and the challenges.
00:01:21 --> 00:01:23
She wanted to go down the aisle
in flowers -
00:01:23 --> 00:01:26
oh, my God - and come back in
a diamond tiara.
00:01:26 --> 00:01:28
That is one of the tallest orders
00:01:28 --> 00:01:31
that any woman could ever have given
her hairdresser!
00:01:31 --> 00:01:34
This two-part series tells the inside story
00:01:34 --> 00:01:37
of a century of Britain's Royal weddings.
00:01:48 --> 00:01:51
On Thursday 26th April 1923,
00:01:51 --> 00:01:51
a young Scottish aristocrat,
00:01:53 --> 00:01:53
Lady Elizabeth Bowes-Lyon, later the
QueenMother,
00:01:56 --> 00:01:57
married King George V's second
son, Prince Albert.
00:02:00 --> 00:02:04
The day was captured on Pathe' newsreel.
```

Figure 6.2: ASR (left) vs subtitles (right) for a documentary of Britain's Royal Weddings from the BBC. The ASR outputs contain a lot of errors which might be corrected with the help of vision, and multimodal models.

- **Classical alignment-style problems**: From the previous example, it is clear that the multimodal framework proposed here has opened up possibilities for a variety of classical alignment-style problems. One important class of these alignment problems is *the mapping of names and faces*. Earlier works such as that of Pham et al. [70] have attempted to view the name-face alignment as a mapping of names to faces or a mapping of faces to names. With our unified approach, we could not only tackle the problem jointly, but also leverage the temporal coherence in video frames, and resolve ambiguous mentions of names. These could also be applied to identify names and faces in live commentaries of events. One line of research lies in applying the multi-modal wildlife recognition framework to other entities such as people.

  As hinted in chapter 5, the methods could be applied to a wider class of concepts, such as furniture or food or generic objects. An interesting line of research is to apply and study our framework on indoor datasets such

as the NYU-Depth V2 dataset [89] to recognize pieces of furniture, or on cooking datasets such as TACoS [78] or MPII [80] to identify entities such as knives, cutting boards, vegetables etc.

- **Beyond video documentaries**: The methods presented in chapters 3 and 4 essentially treat the video as a set of pictures with associated text. These approaches could be applied to label any set of pictures with relevant text, for example, people in news articles, images in encyclopedia, user pins on Pinterest, or even articles on e-commerce websites such as Amazon.com.

- **Beyond subtitles**: An alternative to the subtitles that we used is the output of an Automatic Speech Recognition (ASR) system. ASR outputs typically contain several errors due to absence of sentence boundaries, and the nature of speech in general. Figure 6.2 shows a comparison of the ASR outputs [81] and the subtitles of the BBC video documentary titled Britain's Royal Weddings[1]. Note that there are a lot of errors in the ASR output, for example, '*in a diamond tiara*' is transcribed as '*and AT&T*'. ASR might benefit from a knowledge of what is pictured on the vision side. It would be interesting to apply our multi-modal framework for both recognizing entities or actions on the vision side, and for correcting and improving ASR data at least at a word or phrase level (e.g., by correcting '*and AT&T*' into the correct phrase '*in a diamond tiara*'), if not at a sentence or discourse level.

[1] http://www.bbc.co.uk/programmes/b010ptz1 (accessed May 15, 2017).

Appendix A

Metrics for Evaluating the Entity Linking on Text

We denote a set of mentions referring to the same entity as an entity cluster. Given a set of key (ground-truth) entity clusters K, and a set of response (system-generated) entity clusters R, with each entity cluster comprising one or more mentions, each metric generates its variation of a precision and recall measure. The MUC measure is the oldest and most widely used. It focuses on the *links* (or pairs of mentions) in the data. The number of common links between elements in K and R divided by the number of links in K represents the recall, whereas, precision is the number of common links between elements in K and R divided by the number of links in R. This metric prefers systems that have more mentions per cluster; a system that creates a single cluster of all the mentions will get a 100% recall without significant degradation in its precision. It ignores recall for singleton clusters, or entities with only one mention.

The B^3 metric tries to addresses MUC's shortcomings, by focusing on the *mentions* and computes recall and precision scores for each mention. If K is the key entity cluster containing mention m, and R is the response entity cluster containing mention m, then recall for the mention m is computed as $\frac{|\mathsf{K} \cap \mathsf{R}|}{|\mathsf{K}|}$ and precision for the same is is computed as $\frac{|\mathsf{K} \cap \mathsf{R}|}{|\mathsf{R}|}$. Overall recall and precision are the average of the individual mention scores.

CEAF aligns every response cluster with at most one key cluster by finding the best one-to-one mapping between the clusters using an entity similarity metric. This is a maximum bipartite matching problem solved by the Kuhn-Munkres

algorithm. This metric works at the level of the *entity* cluster. Depending on the similarity, there are two variations: a) *entity* based CEAF - CEAF_e and b) *mention* based CEAF - CEAF_m. Recall is the total similarity divided by the number of mentions in K, and precision is the total similarity divided by the number of mentions in R. In this work, we use CEAF_e for evaluation, similar to the state-of-the-art coreference resolution and entity linking systems [21, 54, 22].

Bibliography

[1] Afkham, H. M., Targhi, A. T., Eklundh, J.-O., and Pronobis, A. Joint visual vocabulary for animal classification. In *Proceedings of the 19th International Conference on Pattern Recognition* (2008), IEEE, pp. 1–4.

[2] Alfonseca, E., and Manandhar, S. An unsupervised method for general named entity recognition and automated concept discovery. In *Proceedings of the 1st International Conference on General WordNet, Mysore, India* (2002), pp. 34–43.

[3] Bagga, A., and Baldwin, B. Algorithms for scoring coreference chains. In *Proceedings of the First International Conference on Language Resources and Evaluation Workshop on Linguistics Coreference* (1998), vol. 1, Citeseer, pp. 563–566.

[4] Ben-Gal, I. Bayesian networks. *Encyclopedia of statistics in quality and reliability* (2007).

[5] Berg, T. L., Berg, A. C., Edwards, J., Maire, M., White, R., Teh, Y. W., Learned-Miller, E. G., and Forsyth, D. A. Names and faces in the news. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2004), vol. 2, pp. 848–854.

[6] Berg, T. L., and Forsyth, D. A. Animals on the Web. In *Proceedings of the IEEE Conference Computer Vision and Pattern Recognition* (2006), vol. 2, IEEE, pp. 1463–1470.

[7] Bergamo, A., and Torresani, L. Exploiting weakly-labeled Web images to improve object classification: A domain adaptation approach. In *Proceedings of the Advances in Neural Information Processing Systems* (2010), pp. 181–189.

[8] Biber, D. *Variation across Speech and Writing*. Cambridge University Press, 1991.

[9] Bishop, C. Pattern Recognition and Machine Learning (Information Science and Statistics). *Springer, New York* (2007).

[10] Brunstein, A. Annotation guidelines for answer types. *LDC2005T33, Linguistic Data Consortium, Philadelphia* (2002).

[11] Bulkin, D. A., and Groh, J. M. Seeing sounds: visual and auditory interactions in the brain. *Current Opinion in Neurobiology 16*, 4 (2006), 415–419.

[12] Bunescu, R. C., and Pasca, M. Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics (EACL)* (2006), vol. 6, pp. 9–16.

[13] Burghardt, T., and Campbell, N. Generic phase curl localisation for an individual identification of turing-patterned animals. *Visual Observation and Analysis of Animal and Insect Behavior* (2010), 17–21.

[14] Chatfield, K., Simonyan, K., Vedaldi, A., and Zisserman, A. Return of the devil in the details: Delving deep into convolutional nets. In *Proceedings of the British Machine Vision Conference* (2014).

[15] Coates-Stephens, S. The analysis and acquisition of proper names for the understanding of free text. *Computers and the Humanities 26*, 5-6 (1992), 441–456.

[16] Coughlan, J. M., and Ferreira, S. J. Finding deformable shapes using loopy belief propagation. In *Proceedings of the European Conference on Computer Vision* (2002), Springer, pp. 453–468.

[17] Daumé III, H. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics* (2007), pp. 256–263.

[18] Dempster, A. P., Laird, N. M., and Rubin, D. B. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (methodological)* (1977), 1–38.

[19] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2009), IEEE, pp. 248–255.

[20] Dougherty, J., Ron, K., and Mehran, S. Supervised and unsupervised discretization of continuous features. In *Proceedings of the Twelfth International Conference on Machine Learning* (1995), vol. 12, San Francisco, CA: Morgan Kaufmann, pp. 194–202.

[21] Durrett, G., and Klein, D. Easy victories and uphill battles in coreference resolution. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (Seattle, Washington, October 2013), Association for Computational Linguistics.

[22] Durrett, G., and Klein, D. A joint model for entity analysis: Coreference, typing, and linking. In *Transactions of the Association for Computational Linguistics* (2014).

[23] Dusart, T., Nurani Venkitasubramanian, A., and Moens, M.-F. Cross-modal alignment for wildlife recognition. In *Proceedings of the 2nd ACM International Workshop on Multimedia Analysis for Ecological Data* (2013), ACM, pp. 9–14.

[24] Ernst, A., and Kublbeck, C. Fast face detection and species classification of african great apes. In *Proceedings of the 8th IEEE International Conference on Advanced Video and Signal-Based Surveillance* (2011), IEEE, pp. 279–284.

[25] Everingham, M., Sivic, J., and Zisserman, A. "hello! My name is... Buffy"–automatic naming of characters in TV video. In *Proceedings of the British Machine Vision Conference* (2006), vol. 2, p. 6.

[26] Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., and Zisserman, A. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[27] Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., and Lin, C.-J. Liblinear: A library for large linear classification. *Journal of Machine Learning Research 9* (2008), 1871–1874.

[28] Fang, H., Gupta, S., Iandola, F., Srivastava, R. K., Deng, L., Dollár, P., Gao, J., He, X., Mitchell, M., Platt, J. C., et al. From captions to visual concepts and back. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 1473–1482.

[29] Fernando, B., Habrard, A., Sebban, M., and Tuytelaars, T. Unsupervised visual domain adaptation using subspace alignment. In *Proceedings of the IEEE International Conference on Computer Vision* (2013), pp. 2960–2967.

[30] Ferreira, F., and Tanenhaus, M. K. Introduction to the special issue on language–vision interactions. *Journal of Memory and Language 57*, 4 (2007), 455–459.

[31] Frassinetti, F., Bolognini, N., and Làdavas, E. Enhancement of visual perception by crossmodal visuo-auditory interaction. *Experimental Brain Research 147*, 3 (2002), 332–343.

[32] Frey, B. J., and MacKay, D. J. A revolution: Belief propagation in graphs with cycles. In *Proceedings of Advances in Neural Information Processing Systems* (1998), Morgan Kaufmann Publishers, pp. 479–485.

[33] Girshick, R., Donahue, J., Darrell, T., and Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2014), pp. 580–587.

[34] Gomez, A., and Salazar, A. Towards automatic wild animal monitoring: Identification of animal species in camera-trap images using very deep convolutional neural networks. *arXiv preprint arXiv:1603.06169* (2016).

[35] Gopalan, R., Li, R., and Chellappa, R. Domain adaptation for object recognition: An unsupervised approach. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (2011), IEEE, pp. 999–1006.

[36] Guadarrama, S., Krishnamoorthy, N., Malkarnenkar, G., Venugopalan, S., Mooney, R., Darrell, T., and Saenko, K. Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (2013), IEEE, pp. 2712–2719.

[37] Guillaumin, M., Mensink, T., Verbeek, J., and Schmid, C. Automatic face naming with caption-based supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2008), IEEE, pp. 1–8.

[38] Hachey, B., Radford, W., Nothman, J., Honnibal, M., and Curran, J. R. Evaluating entity linking with Wikipedia. *Artificial Intelligence 194* (2013), 130–150.

[39] Hariharan, B., and Girshick, R. Low-shot visual object recognition. *arXiv preprint arXiv:1606.02819* (2016).

[40] Hellier, P., Demoulin, V., Oisel, L., and Perez, P. A contrario shot detection. In *Proceedings of the 19th IEEE International Conference on Image Processing (ICIP)* (2012), IEEE, pp. 3085–3088.

[41] Holmberg, J., Norman, B., and Arzoumanian, Z. Estimating population size, structure, and residency time for whale sharks rhincodon typus through collaborative photo-identification. *Endangered Species Research 7*, 1 (2009), 39–53.

[42] Javed, O., Ali, S., and Shah, M. Online detection and classification of moving objects using progressively improving detectors. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)* (2005), vol. 1, IEEE, pp. 696–701.

[43] Jurafsky, D., and James, H. Speech and Language Processing: an Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition. *Pearson Education,* (2000).

[44] Karpathy, A., and Fei-Fei, L. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2015), pp. 3128–3137.

[45] Kazemzadeh, S., Ordonez, V., Matten, M., and Berg, T. L. Referit game: Referring to objects in photographs of natural scenes. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2014), pp. 787–798.

[46] Khosla, A., Jayadevaprakash, N., Yao, B., and Li, F.-f. L.: Novel dataset for fine-grained image categorization. In *Proceedings of First Workshop on Fine-Grained Visual Categorization, IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2011), Citeseer.

[47] Kindermann, R., and Snell, L. *Markov Random Fields and their Applications*. American Mathematical Society, 1980.

[48] Koehn, P., and Schroeder, J. Experiments in domain adaptation for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation* (2007), Association for Computational Linguistics, pp. 224–227.

[49] Koller, D., and Friedman, N. *Probabilistic Graphical Models: Principles and Techniques*. MIT press, 2009.

[50] Kong, C., Lin, D., Bansal, M., Urtasun, R., and Fidler, S. What are you talking about? Text-to-image coreference. In *Proceedings of the*

IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2014), IEEE, pp. 3558–3565.

[51] Lai, K., Bo, L., Ren, X., and Fox, D. RGB-D object recognition: Features, algorithms, and a large scale benchmark. In *Consumer Depth Cameras for Computer Vision*. Springer, 2013, pp. 167–192.

[52] Lampert, C. H., Nickisch, H., and Harmeling, S. Learning to detect unseen object classes by between-class attribute transfer. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2009), IEEE, pp. 951–958.

[53] LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. Backpropagation applied to handwritten zip code recognition. *Neural Computation 1*, 4 (1989), 541–551.

[54] Lee, H., Chang, A., Peirsman, Y., Chambers, N., Surdeanu, M., and Jurafsky, D. Deterministic coreference resolution based on entity-centric, precision-ranked rules. *Computational Linguistics 39*, 4 (2013), 885–916.

[55] Leibe, B., Cornelis, N., Cornelis, K., and Van Gool, L. Dynamic 3d scene analysis from a moving vehicle. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2007), IEEE, pp. 1–8.

[56] Leser, U., and Hakenberg, J. What makes a gene name? named entity recognition in the biomedical literature. *Briefings in Bioinformatics 6*, 4 (2005), 357–369.

[57] Li, Y., Liu, L., Shen, C., and van den Hengel, A. Mid-level deep pattern mining. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 971–980.

[58] Liu, X., Li, Y., Wu, H., Zhou, M., Wei, F., and Lu, Y. Entity linking for tweets. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)* (2013), pp. 1304–1311.

[59] Loos, A., and Pfitzer, M. Towards automated visual identification of primates using face recognition. In *Proceedings of the 19th International Conference on Systems, Signals and Image Processing (IWSSIP)* (2012), IEEE, pp. 425–428.

[60] Luo, X. On coreference resolution performance metrics. In *Proceedings of the Conference on Human Language Technology and Empirical Methods*

in Natural Language Processing (2005), Association for Computational Linguistics, pp. 25–32.

[61] McCulloch, C. E. The EM algorithm and its extensions. *Journal of the American Statistical Association 93*, 441 (1998), 403–405.

[62] Meredith, M. A., and Stein, B. E. Interactions among converging sensory inputs in the superior colliculus. *Science 221*, 4608 (1983), 389–391.

[63] Miller, G. A. WordNet: A lexical database for English. *Communications of the ACM 38* (1995), 39–41.

[64] Milosavljevic, M., Delort, J.-Y., Hachey, B., Arunasalam, B., Radford, W., and Curran, J. R. Automating financial surveillance. In *International Conference on User Centric Media* (2009), Springer, pp. 305–311.

[65] Nadeau, D., and Sekine, S. A survey of named entity recognition and classification. *Lingvisticae Investigationes 30*, 1 (2007), 3–26.

[66] Nigam, K., McCallum, A. K., Thrun, S., and Mitchell, T. Text classification from labeled and unlabeled documents using EM. *Machine Learning 39*, 2-3 (2000), 103–134.

[67] Olofsson, P., and Andersson, M. *Probability, Statistics, and Stochastic Processes*. John Wiley & Sons, 2012.

[68] Parker, A. In the blink of an eye: How vision kick-started the big bang of evolution. *Simon and Shuster* (2003).

[69] Pearl, J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 2014.

[70] Pham, P. T., Moens, M.-F., and Tuytelaars, T. Cross-media alignment of names and faces. *IEEE Transactions on Multimedia 12*, 1 (2010), 13–27.

[71] Pham, P. T., Tuytelaars, T., and Moens, M.-F. Naming people in news videos with label propagation. *IEEE Multimedia 18*, 3 (2011), 44–55.

[72] Ponce, J., Hebert, M., Schmid, C., and Zisserman, A. *Toward category-level object recognition*, vol. 4170. Springer, 2007.

[73] Pradhan, S., Luo, X., Recasens, M., Hovy, E. H., Ng, V., and Strube, M. Scoring coreference partitions of predicted mentions: A reference implementation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)* (2014), pp. 30–35.

[74] Pradhan, S., Ramshaw, L., Marcus, M., Palmer, M., Weischedel, R., and Xue, N. CoNLL-2011 shared task: Modeling unrestricted coreference in ontonotes. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning: Shared Task* (2011), Association for Computational Linguistics, pp. 1–27.

[75] Ramanan, D., Forsyth, D. A., and Barnard, K. Building models of animals from video. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 28*, 8 (2006), 1319–1334.

[76] Ramanathan, V., Joulin, A., Liang, P., and Fei-Fei, L. Linking people in videos with "their" names using coreference resolution. In *Proceedings of the European Conference on Computer Vision (ECCV)* (2014), Springer, pp. 95–110.

[77] Razavian, A. S., Azizpour, H., Sullivan, J., and Carlsson, S. CNN features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops* (2014), IEEE, pp. 512–519.

[78] Regneri, M., Rohrbach, M., Wetzel, D., Thater, S., Schiele, B., and Pinkal, M. Grounding action descriptions in videos. *Transactions of the Association for Computational Linguistics (TACL) 1* (2013), 25–36.

[79] Ren, S., He, K., Girshick, R., and Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Proceedings of Advances in Neural Information Processing systems (NIPS)* (2015), pp. 91–99.

[80] Rohrbach, A., Rohrbach, M., Tandon, N., and Schiele, B. A dataset for movie description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2015).

[81] Ronny, R., Shakoor, A., Brugnara, F., and Gretter, R. *The FBK ASR System for Evalita 2011.* Springer, Berlin, Heidelberg, 2013, pp. 295–304.

[82] Roth, D., and Yih, W.-t. Probabilistic reasoning for entity & relation recognition. In *Proceedings of the 19th International Conference on Computational Linguistics-Volume 1* (2002), Association for Computational Linguistics, pp. 1–7.

[83] Saenko, K., Kulis, B., Fritz, M., and Darrell, T. Adapting visual category models to new domains. *Computer Vision–ECCV 2010* (2010), 213–226.

[84] Schmid, C. Constructing models for content-based image retrieval. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)* (2001), vol. 2, IEEE, pp. II–39.

[85] Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Networks 61* (2015), 85–117.

[86] Schmidt, M. UGM: Matlab code for undirected graphical models, 2012. *URL* `http://www.cs.ubc.ca/~schmidtm/Software/UGM.html`.

[87] Seitner, J., Bizer, C., Eckert, K., Faralli, S., Meusel, R., Paulheim, H., and Ponzetto, S. A large database of hypernymy relations extracted from the Web. In *Proceedings of the 10th Edition of the Language Resources and Evaluation Conference, Portoroz, Slovenia* (2016).

[88] Shen, W., Wang, J., and Han, J. Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Transactions on Knowledge and Data Engineering 27*, 2 (2015), 443–460.

[89] Silberman, N., Hoiem, D., Kohli, P., and Fergus, R. Indoor segmentation and support inference from RGB-D images. In *Proceedings of the European Conference on Computer Vision (ECCV)* (2012), Springer, pp. 746–760.

[90] Snavely, N., Seitz, S. M., and Szeliski, R. Modeling the world from Internet photo collections. *International Journal of Computer Vision 80*, 2 (2008), 189–210.

[91] Song, D., Qin, N., Xu, Y., Kim, C. Y., Luneau, D., and Goldberg, K. System and algorithms for an autonomous observatory assisting the search for the ivory-billed woodpecker. In *Proceedings of the IEEE International Conference on Automation Science and Engineering* (2008), IEEE, pp. 200–205.

[92] Spector, P. *An Introduction to S and S-PLUS*. Duxbury press: Wadsworth Inc, 1994.

[93] Stoyanov, V., Cardie, C., Gilbert, N., Riloff, E., Buttler, D., and Hysom, D. Coreference resolution with Reconcile. In *Proceedings of the ACL 2010 Conference Short Papers* (2010), Association for Computational Linguistics, pp. 156–161.

[94] Swanson, A., Kosmala, M., Lintott, C., Simpson, R., Smith, A., and Packer, C. Snapshot Serengeti, high-frequency annotated camera trap images of 40 mammalian species in an African savanna. *Scientific Data 2* (2015), 150026.

[95] Szeliski, R. *Computer Vision: Algorithms and Applications*. Springer Science & Business Media, 2010.

[96] Tommasi, T., and Caputo, B. Frustratingly easy nbnn domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision* (2013), pp. 897–904.

[97] Venkitasubramanian, A. N., Tuytelaars, T., and Moens, M.-F. Wildlife recognition in nature documentaries with weak supervision from subtitles and external data. *Pattern Recognition Letters, Elsevier 81* (2016), 63–70.

[98] Vilain, M., Burger, J., Aberdeen, J., Connolly, D., and Hirschman, L. A model-theoretic coreference scoring scheme. In *Proceedings of the 6th Conference on Message Understanding* (1995), Association for Computational Linguistics, pp. 45–52.

[99] Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. The Caltech-UCSD Birds-200-2011 Dataset, Technical Report CNS-TR-2011-001, California Institute of Technology.

[100] Zhang, N., Donahue, J., Girshick, R., and Darrell, T. Part-based R-CNNs for fine-grained category detection. In *Proceedings of the European Conference on Computer Vision (ECCV)* (2014), Springer, pp. 834–849.

Curriculum Vitae

Aparna Nurani Venkitasubramanian was born in Chennai, India. She received a Bachelors in Engineering and Computer Science from Anna University, Chennai, India in 2006. From 2006 to 2010, she worked with Hewlett Packard Inc. as an SAP Technical consultant, and was responsible for end-to-end software development. She obtained a Master of Science in Artificial Intelligence from KU Leuven, Belgium in 2011 and started a Ph.D. program at the LIIR (Language Intelligence and Information Retrieval) research group at the Department of Computer Science, KU Leuven, Belgium. Her research interests lie at intersection of Natural Language Processing and Computer Vision.

List of Publications

Articles in internationally reviewed academic journals

- **Nurani Venkitasubramanian, A.**, Tuytelaars, T., Moens, M.-F. (2017). *Entity linking across vision and language*, Multimedia Tools and Applications, Springer. DOI:10.1007/s11042-017-4732-8.
- **Nurani Venkitasubramanian A.**, Tuytelaars T., Moens M.-F. (2016). *Wildlife recognition in nature documentaries with weak supervision from subtitles and external data.* Pattern Recognition Letters, Elsevier 81, 63-70.

Papers at international scientific conferences and symposia, published in full in proceedings

- **Nurani Venkitasubramanian A.**, Tuytelaars T., Moens M.-F. (2017). *Learning to recognize animals by watching documentaries: using subtitles as weak supervision.* Proceedings of the 6th Workshop on Vision and Language (VL'17) at EACL 2017. Valencia, 4 April 2017.
- Shrestha N., **Nurani Venkitasubramanian A.**, Moens M.-F. (2014). *Key event detection in video using ASR and visual data.* Proceedings of the COLING Workshop on Vision and Language (VL'14). Dublin, Ireland, 23 August 2014 (pp. 46-53).
- Dusart T., **Nurani Venkitasubramanian A.**, Moens M.-F. (2013). *Cross-modal alignment for wildlife recognition.* Proceedings of the 2nd ACM International Workshop on Multimedia Analysis for Ecological Data. Barcelona, Spain, 21 October 2013 (pp. 9-14).

- **Nurani Venkitasubramanian A.**, Moens M.-F. (2013). *Estimating the breadth of search queries.* Proceedings of the 10th International RIAO Conference on Open Research Areas in Information Retrieval. Lisbon, Portugal, 22-24 May 2013 (pp. 109-112). New York: ACM.
- **Nurani Venkitasubramanian A.**, Moens M.-F. (2013). *Selection of search facets.* Proceedings of CORIA 2013 - 10th French Information Retrieval Conference. Neuchâtel, Switzerland, 3-5 April 2013 (pp. 73-84).

Papers at other professionally oriented conferences and symposia, published in full in proceedings

- **Nurani Venkitasubramanian A.**, Moens M. (2013). *Summarization and expansion of search facets.* Proceedings of the Thirteenth Dutch-Belgian Information Retrieval Workshop. Delft, The Netherlands, 26 April 2013, vol.986, 50-51.

FACULTY OF ENGINEERING SCIENCE
DEPARTMENT OF COMPUTER SCIENCE
LANGUAGE INTELLIGENCE AND INFORMATION RETRIEVAL (LIIR)
Celestijnenlaan 200A box 2402
B-3001 Leuven
aparna.venkit@gmail.com